AUGMENTED REALITY
WITH DIGITAL METROLOGY FOR ASSEMBLY

by

Rahma Nawab

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
Optical Science and Engineering

Charlotte

2022

Approved by:

____
Dr. Angel Davies Allen

____
Dr. Edward Morse

____
Dr. Stuart Smith

____
Dr. Kosta Falaggis

ABSTRACT

RAHMA NAWAB. Augmented Reality with Digital Metrology for Assembly

(Under the Direction of DR. ANGELA DAVIES ALLEN)

This dissertation aims to create and demonstrate a fast and inexpensive quantitative dimensional inspection system for industrial assembly line applications that can detect position errors of assembled components on a scale of 1 mm and larger. The researched and developed data acquisition and the computational pipeline is presented. Position error detection of 1 mm and higher is demonstrated on a 40-mm high and 8mm radius post welded onto a 30 mm by 30 mm steel plate.

Data acquisition is performed using an open-source photogrammetry architecture to gather a 3D point cloud of the assembled part. The photogrammetry architecture involves a structure from motion (SfM) pipeline to obtain a sparse point cloud, and a depth map merging method is used to generate a dense point cloud.

The component's position on the assembly is calculated by comparing the point cloud with the CAD model. A method using the iterative closest point (ICP) algorithm establishes a global coordinate system for the data to align with the CAD model. Once the global coordinate system is established and aligned, the position of individual parts is estimated relative to the reference. This method was able to identify a 1-mm and larger position error of a post, as described above, welded onto a steel place.

A partial uncertainty evaluation shows that the position error can be estimated with uncertainty no better than several fractions of a millimeter. Uncertainty contributions can be divided up into three groups: i) random noise and possible bias in the acquired point cloud data,

ii) the actual shape of the component compared to the ideal (CAD model) component shape, and iii) data processing choices. The investigations into these aspects showed that, with enough points in the reference, a down-sampled measured data set, and a low ICP threshold, the limiting factors in the uncertainty come from spatial bias due to 3D reconstruction and object surface roughness. Point-wise random noise in the measured 3D point cloud was also investigated with commercial software (PolyWorks) and open-source algorithms and found to contribute negligibly to the combined uncertainty compared to the uncertainty caused by the object shape and surface texture. These effects require more work to estimate a comprehensive combined uncertainty.

The pipeline was made user-friendly by creating an augmented reality (AR) application. This application detected the component of interest in a live video feed and overlayed the position error information. Object detection was done by creating guide views of the CAD model.

ACKNOWLEDGMENTS

I would like to acknowledge and give my warmest thanks to my supervisor Dr. Angela Allen who made this work possible. Her guidance and advice carried me through all the stages of researching and writing my dissertation. I would also like to thank Dr. Morse, Dr. Smith, Dr. Falaggis, Dr. Farahi, and Dr. Lee for their brilliant suggestions, use of their labs, and help throughout my journey at UNCC. Additionally, a huge thanks to my parents, Farah and Muzammil, and family Wahib, Shafaq, Hassan, and Hammad, for their unwavering support. Lastly, I would like to acknowledge my friends for their encouragement.

DEDICATION

This work is dedicated to my parents, Dr. Muzammil and Dr. Farah, who have been a constant source of support, love, and encouragement during all the phases of my life, from pre-school to graduate school. Thank you for all your sacrifices and excellent examples to live up to.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CAD	Computer Aided Design
AR	Augmented Reality
ISO	International Organization for Standards
CMM	Coordinate Measuring Machine
SIFT	Scale Invariant Feature Transform
SURF	Speeded up Robust Transform
LIDAR	Light Detection and Radar
SfM	Structure from Motion
ICP	Iterative Closest Point
DOF	Degrees of Freedom

CHAPTER 1 INTRODUCTION

1-1 MOTIVATION AND GOALS

This dissertation was motivated by conversations with industrial affiliates who wanted to develop a prototype model for industrial quality inspection in assembly. The proposed goal is a cost-effective, accessible, and user-friendly solution to millimeter-scale positioning errors in the assembly stage of a manufacturing process. Examples of desired measurements include the position of a boss or bolt on cars and aircraft manifolds during assembly. Industrial environments vary greatly in location, size, and facilities available; owing to this, a large majority of applications would be an assembly of large structures, which are difficult to inspect in a traditional metrology lab.

It is proposed that the measurement be photogrammetry based, using smartphones or tablets for the data acquisition of images for the measurement. These devices acquire high-quality images in varying industrial environments, are economical, and are available in most locations. After 3D point cloud reconstruction of the part through photogrammetry, the computer-aided design (CAD) models of the part under assembly would then be used to compare positioning errors of components. The final step of the pipeline would focus on a user-friendly interface to automate the inspection process and deliver the measurement result using augmented reality (AR).

The goal of this dissertation is to advance the literature in these three areas specific to the needs of this application and combine the steps into one cohesive pipeline that performs fast and economic quantitative position information for industrial assembly line applications. To realize this goal, we designed, manufactured, and evaluated an experimental artifact to test the pipeline. This dissertation explains the research and development activities for each pipeline step, divided into chapters. The chapters explain the approach, design, and evaluation for each part of the project.

While creating this quality inspection pipeline, many challenges were addressed—the project involved linking together different independent pieces; each requiring significant research and development. The first task was to create a photogrammetry architecture using open-source libraries in C++ and python. The main challenge here involved testing different image processing and reconstruction algorithms to optimize the 3-D point cloud acquisition for this metrology goal. The second part involved point cloud registration and CAD model comparison to estimate a measurand that is the position error in the assembly. We tested several algorithms and platforms, both custom code and commercial software, to find our application's most reliable and efficient method. This dissertation also explores the impact of several processing parameter choices on the measurement uncertainty and the effects of point-wise random noise on the final position-error measurand.

A literature review shows that a cohesive package, as proposed here, does not exist in the industrial assembly domain that measures and utilizes this quantitative data to display information in the augmented reality platform for quality inspection during assembly. The primary goal of this dissertation is to bridge the gap between metrology and user experience with a prototype model that performs large-scale close-range dimensional metrology in the manufacturing assembly industry with an augmented reality interface.

1-2 LITERATURE SURVEY

This literature survey encompasses the challenges faced by the quality inspection industry. Data acquisition is the first significant component of the pipeline, and we explore the different avenues available, their pros and cons for our application, and the data quality. Dimensional measurements in an assembly are split into contact-based and optical-based methods. Our focus during this literature review is on the latter. The survey also touches on the importance of augmented reality within the quality inspection industry, aiding the user through a fast phone/tablet and AR-based inspection process.

1-2-1 QUALITY INSPECTION IN INDUSTRY - CHALLENGES

Quality inspection in manufacturing characterizes how well a product's characteristics agree with its intended design. The process requires measuring, assessing, and testing the part. Quality inspection is a crucial step in manufacturing as it eliminates the distribution of faulty parts, saves overhead costs, and identifies faults in the manufacturing process.

Quality inspection constitutes a significant fraction of industrial manufacturing activities. In the '80s, as estimated by Miller [34], inspection-related tasks utilized 10 percent of the workforce; for more dangerous applications like aircraft manufacturing, quality control accounts for around 60 percent of the total manufacturing cost [34].

In many sectors, inspection traditionally relies on human vision for the assembly stage of manufacturing, where the part is qualitatively compared with a sketch or a CAD model and then approved or discarded. According to a recent study, human errors accounted for 23% of the measurements being done incorrectly while performing visual quality inspection [1].

It has been shown that a device or measurement system operator always influences the manufacturing process while performing measurement and characterization [37]. Specifically, when the setup of user A differs from user B, measurements will vary because of operator error. These errors can be reduced by defining and following a standard 'good practice', such as verification procedures described by ISO 10360 standards [37]; however, measurement uncertainty will still be limited by the skill of the instrument handler. This calls for an automated and machine-reliant instrument to perform the task. Independent measurements and analysis are thus required to optimize the inspection process without reliance on an experienced operator [37].

In recent decades, manufacturing industries have seen a digital transformation within a particular task and as part of a more extensive pipeline [34]. In the age of knowledge sharing, digital technology has made it possible to link systems throughout the production process [34]. Technological advances have invented more complex and efficient systems to perform quality inspection. The inspection process for assembly applications can thus be divided into three main components:

1. data acquisition or measurement of the object in question,
2. analysis and evaluation of the data to pass quality check, and
3. reporting the findings to the user.

Ideally, acquiring the dimensional data of the object of interest on the industry floor must be fast and economical. This need inevitably compromises uncertainty, time, and equipment cost. Analysis and measurement evaluation is usually done by comparison with a reference, and the findings are typically displayed to the user in a table format.

1-2-2 DATA ACQUISITION TECHNIQUES

While automating the inspection process, the first step is often the dimensional measurement of the object under evaluation. The measurements can be divided into two categories: contact and non-contact. The former generally uses a stylus in a coordinate measuring machine, and the latter typically acquires this dimensional information with light-based techniques such as laser scanners, photogrammetry, structured light, and laser triangulation [2].

Coordinate measuring machines offer high accuracy and flexibility in metrology, making them a popular choice in the manufacturing industry [32]. The contact CMM, though very accurate and extensively used in industry, has limitations for monitoring parts in the assembly stage. In addition to the expense, the contact force may damage the surface, as can happen with polymer parts. Furthermore, these machines are also slow because of the serial nature of the measurements [6]. Many applications do not call for CMM-level measurement uncertainty, meaning the assembly tolerances are often relaxed compared to individual component dimensional tolerances. We aim to develop a fast and inexpensive dimensional measurement method for assembly applications with millimeter-scale positioning tolerances and submillimeter measurement uncertainties.

1-2-2-1 NON-CONTACT MEASUREMENTS OVERVIEW

For many manufacturing processes, the productivity rate is limited by the measurement process at the final assembly quality inspection step. Thus, productivity is improved by reducing the time investment in the assembly evaluation step. Optical instruments tend to be image-based which means data-acquisition involves detector arrays and parallel data acquisition, thus measurement times are faster than serial contact-based instruments like the CMM. Optical inspection also eliminates the risk of contact-related damage. Optical techniques collect high-density data and can offer more flexibility in measuring complex features and shapes [33].

In recent years, with the advent of fast computing powers and devices, metrology in the manufacturing industry can now be used as an in-line inspection step while performing quality control. Conventionally used as a final step in product manufacturing, advances in industry have now enabled supervising and diagnostic data to be available in real-time throughout the process to facilitate quality inspection and data analysis [35, 36]. Conventional probe-based coordinate measurement techniques are relatively slow for a small number of data points, making contact measurement instruments an impractical solution for integrated metrology in quality inspection [43].

Optical measurement technologies are easier to apply to integrated quality inspection processes in manufacturing, as they are non-contact and acquire high-density data at a higher rate. It is common for manufacturing industries to have several inter-connected devices that collect and share data in real-time, making non-contact metrology processes an ideal measurement system as they offer automated and quick verification solutions. In conjunction with image processing techniques, vision systems are being utilized to store quality information inside the production environment. Similarly, inspection instruments for measuring geometrical and coordinate features in aerospace and automotive inspections utilize optical technologies, particularly laser-based devices [37].

A thorough review of optical measurement techniques in industrial environments [59] explored different techniques, such as laser scanners, photogrammetry, structured light, and laser triangulation. A comparative review of the study is presented in Table 1 [2].

Table 1: Comparison of optical techniques [2].

	Accuracy	Range	Weight	Safety	Processing Time	Environmental Influences
Photogrammetry	✓ (50 µm)	✓	✓	✓	× (Image Processing)	× (brightness)
Projected Texture Stereo Vision	✓ (0.1 mm)	× (0.25 – 3 m)	✓	✓	× (Image processing)	× (brightness)
Time of Flight	× (10 mm)	× (0.8 - 8 m)	✓	✓	✓	✓
Structured White Light	✓ (0.127 mm)	✓	× (projector)	✓	× (Remain static)	× (Light, brightness)
Structured Blue LED Light	✓ (34 µm)	× (157 – 480 mm)	✓	✓	× (Remain static)	✓
Light Coding	× (10 mm)	× (1 – 3 m)	✓	✓	✓	× (sun)
Laser Triangulation	✓	✓	✓	× (Laser power)	✓	× (brightness)

Photogrammetry is a machine vision-based technique that uses triangulation to obtain 3D point clouds from 2-D images. The images taken at different vantage points are used to obtain 3D data based on the shifts in image features. The main applications of photogrammetry include 6 degrees of freedom measurements, such as object tracking and robot navigation systems. As shown in Table 1, the error in photogrammetry measurements can be as low as 50 microns using a Nikon D3 camera, coded targets, and following other best practices [127]. This technique requires cameras, so it does not have weight or safety issues; however, it is prone to changes in measurements due to lighting conditions. Furthermore, the data processing step takes time.

Time of flight measurement utilizes light pulses, which are directed at the object to be measured. A sensor captures the reflected light, and the distance between the sensor and object is calculated by measuring time delay. The width of the light pulse is used to determine the measurement range. Swissranger 4K offers a measurement range from 0.8 m to 8 m [128] and low measurement accuracy.

Structured light measurements use a projector to project structured patterns onto the object to be measured; the distorted pattern due to the object geometry is then captured. Once the system is calibrated, the 3D coordinates of the point cloud can be reconstructed. Among these, blue LED structured techniques offer low measurement uncertainty, with sensors such as HDI 109 from LMI technologies able to measure within a reported uncertainty of 34 µm [2]. The downfall of blue LED measurement techniques is the short working distance and equipment expense used to project the structured light.

In laser triangulation measurements, the angle of the camera and the distance between a camera and laser emitter is used to locate the location of the laser dot on the object and hence the object's 3D coordinates. Laser techniques are often used for scanning, are expensive, and come with safety considerations. The geometric information acquired using laser scanners is accurate with low measurement uncertainty but costly because of expensive scanning equipment and the need for skilled operators [3]. In addition, time and money are needed for data analysis and processing using specialized software.

Structured light techniques are less expensive but have challenges. Projected light visibility can be a problem under certain ambient light conditions. Surface colors must be considered and can directly impact the projected light pattern visibility.

1-2-2-2 NON-CONTACT MEASUREMENTS BASED ON MACHINE VISION

Among optical measurement techniques, camera-based approaches have emerged as a promising alternative to standard contact sensors as they offer significant advantages such as low cost, ease of setup, and ease of operation. A single video or a few images are enough to calculate dimensional information, offering industrial environmental flexibility [K]. Physical connections to the product and power supplies are not needed for a camera, thus also offering remote location advantages at a low cost with time savings.

Compared to laser measurement devices, which need to be placed reasonably close to the target for laser power or safety concerns, vision-based sensors can be placed a few meters away and still achieve reasonable measurement uncertainty for many applications. Furthermore, the data captured is a parallel acquisition of multiple points compared to the point-wise measurement of conventional serial sensors [42].

Computer vision devices, particularly photogrammetry, use triangulation-based algorithms for 3-D reconstruction. They utilize bundle adjustment to calculate the points in 3-D space and the camera parameters by minimizing the reprojection errors in the image planes. This minimization problem is carried out with linear algebra and is very efficient [40]. Some of the drawbacks, such as occlusions and exposure time, can be overcome with additional light sources to reduce shadows, make geometric features more accessible, and decrease the exposure time [40, 38].

Similarly, much work has been done by utilizing feature information to estimate missing data. Recent work by Sjödahl [39] describes a close-range photogrammetry-based device for in-process inspection of metal sheets. The CAD model of the part is used to detect the target's geometrical features and used to facilitate the reconstruction process, making real-time measurements on the conveyer belt possible at a frequency of 0.5 Hz.

1-2-3 ERROR VISUALISATION THROUGH AUGMENTED REALITY

Assembly quality inspection commonly involves commercial CAD systems, where 3-D CAD models of the components to be assembled are displayed on a screen next to the assembly line to allow for visual identification of errors. The use of CAD software presents limitations, as the user is not alerted about the defect or positional error automatically and needs to look at a color map or measurement data, which can prove to be cumbersome. Augmented reality (AR) can overcome this limitation and unlike virtual reality (VR), AR does not require the entire real world to be modeled, eliminating the high cost of creating an immersive virtual reality environment [11].

Most current AR applications in the industry focus on manual assembly processes, where AR is used to overlay instructions to the user onto the assembly scene. Qualitative quality inspection, training activities, and machining setups are other application examples where AR assists the user in visualizing and detecting defects more efficiently [12].

Developing these AR applications can be hardware and software intensive, and a variety of hardware devices are available. Head-mounted display (HMD) devices are the most popular, accounting for 40 % of all visualization devices in industrial AR applications [12]. Wearing these can, however, be uncomfortable and cause dizziness. Handheld devices and smartphones provide a good alternative because they are cost-effective and easily accessible. Fraunhofer IGD [13] and FARO Visual Inspect [14] have developed commercial devices that offer quality inspection for industries similar to what is described here. The inspection in these systems is qualitative, however, and does not report quantitative errors with uncertainty which is the goal of this project.

1-3 NOVEL PIPELINE PRESENTED IN THIS DISSERTATION

The pipeline in this dissertation provides a system to perform fast and inexpensive *quantitative* quality inspection in an industrial assembly line using dimensional metrology. Dimensional metrology is the science of geometrical measurement, and our focus is on the 1-mm and larger position errors of assembled pieces on a larger part as it is assembled.

We propose a photogrammetry-based approach to deliver high-density 3D point cloud data over extensive object volumes at a low cost without the use of targets [7]. Photogrammetry can be performed using targets, patterns, or a simple dot on a paper with a contrasting color. Although this improves accuracy, placing targets on the scene in an industrial environment is cumbersome. We avoid this step and use only feature-based photogrammetry.

Industrial assembly features often consist of highly textured surfaces with contrasting colors and features, making it an ideal application of *feature-based detection* for photogrammetry without the need for coded or non-coded targets. Using coded targets reduces measurement uncertainty, but feature detection alone is often sufficient for assembly-line positioning error detection on the millimeter or larger scale, as demonstrated here.

Feature detection algorithms in photogrammetry use edge-finding and other feature-specific strategies based on the difference of Gaussians or using a Hessian matrix-based blob detection in popular algorithms like scale-invariant feature transform (SIFT) and the speeded-up robust features (SURF) algorithm. These algorithms rely on the visual properties of the surface and show significantly less noise for these types of surfaces than would result from light detection and ranging (LIDAR)-based sensors [10]. We use algorithms available in the open source, feature-based photogrammetry library OpenMVG as mentioned in Chapter 2, which is flexible and economical.

Point cloud data needs to be visualized and processed for quality inspection for metrology applications. Numerous platforms have been developed for point cloud visualization and processing. They include commercial software packages developed by manufacturers of point cloud measuring devices and independent commercial software packages. GOM Inspect, FARO scene, Polyworks, and PointCab are examples [7]. These software packages offer advanced point cloud processing and error measurement capabilities but are expensive. We propose a simple technique that utilizes the popular point cloud fine registration algorithm, iterative closest point (ICP), and MATLAB to compare our point cloud with the CAD model.

Figure 1 shows the proposed pipeline to perform a quantitative assembly-line inspection. Photogrammetry is combined with a CAD model as a reference for comparison purposes. We then create an AR application to provide an automated quantitative quality inspection platform for assembly-line applications. Point cloud registration is used to calculate a transformation matrix, which is used to calculate the translation and rotation between the two-point clouds and hence the positioning errors between the reference and measured object.

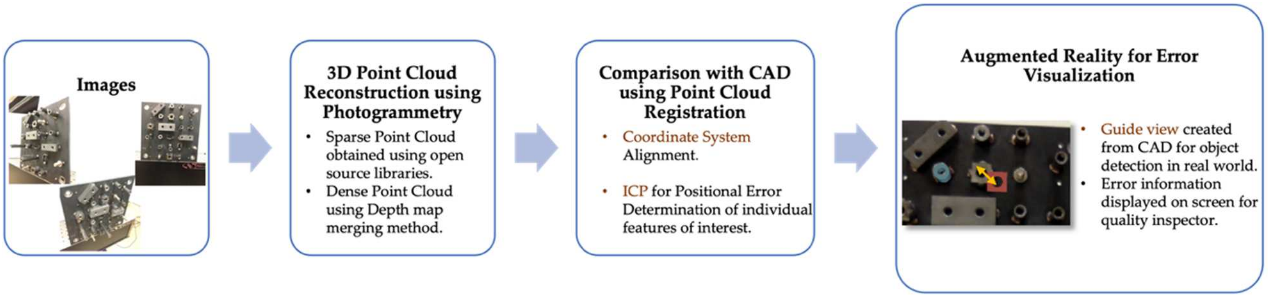

Images
3D Point Cloud Reconstruction using Photogrammetry
• Sparse Point Cloud obtained using open source libraries.
• Dense Point Cloud using Depth map merging method.
Comparison with CAD using Point Cloud Registration
• Coordinate System Alignment.
• ICP for Positional Error Determination of individual features of interest.
Augmented Reality for Error Visualization
• Guide view created from CAD for object detection in real world.
• Error information displayed on screen for quality inspector.

Figure 1. Flowchart of the proposed method**.**

Following is an overview of the research:

1. Artifact design and fabrication to simulate commonly used industrial assembly line features and dimensions. We target the automotive and aircraft assembly applications, where the position of designed features such as a bolt and post are explored.
2. 3-D shape measurement of the component requires constructing a camera-based photogrammetry architecture without targets and uses point cloud reconstruction using open-source libraries.
3. Component position errors in the assembly line are estimated by comparing the point cloud obtained from photogrammetry with the CAD model of the assembled part. The CAD

model is sampled into a point cloud data set and registered with the photogrammetry data. This registration is a two-step process involving coarse and fine registration. Both commercial software and MATLAB are explored to perform this step.

4. Uncertainty studies are done, where the effect of random noise on the measurement is investigated, and the factors limiting the uncertainty of the measurement are explored.
5. Once the position error is estimated, an augmented reality platform displays the quantitative position error information in real-time on a live video feed of the part.

The following four chapters discuss the different steps of the proposed solution to the quantitative inspection problem. Chapter 2 goes over data acquisition using photogrammetry, where we explain the cost-effective avenues available and our photogrammetry architecture. Chapter 3 discusses the CAD comparison, where CAD data is used as a sampled reference point cloud to estimate the position error. Potential uncertainty contributions are studied in Chapter 4. Chapter 5 presents the final step, wherein we implement AR to overlay the quantitative position error information onto a live video feed for the user to visualize and perform the quality check.

CHAPTER 2 DATA ACQUISITION USING PHOTOGRAMMETRY

Our goal in the data acquisition process is to obtain a 3D data set describing the dimensions of the assembly under inspection. The aim is to detect millimeter-level position deviations of assembled components as compared to the reference CAD model of the assembled system. Since we aim at assembly errors in the automotive and aircraft industry, our inspection area will range on the scale of fractions of a meter to several meters, but the methodology could be extended to smaller and larger scales.

Current methods of 3D data acquisition are laser scanners, structured light techniques, machine vision methods, and the coordinate measuring machine. We discuss these methods in more detail in the following sections and discuss the advantages and disadvantages of using photogrammetry for our application. The required density of the 3D data will depend on the component whose assembled position needs to be evaluated. In our case, we will be reconstructing components, such as nuts and bolts, that are approximately 4 cm high and 1 cm wide welded onto a steel plate.

Photogrammetry is a Greek word derived from the Greek word for light, called 'phos', 'gramma' meaning a drawing, and 'meterin' from the greek word meaning measure [44]. As described by the name, photogrammetry is the art of obtaining information about the environment and objects through measuring and interpreting photographic images or electromagnetic patterns [46]. The American Society for Photogrammetry and Remote Sensing defines it as "the art, science, and technology of obtaining reliable information about physical objects and the environment through processes of recording, measuring, and interpreting images and patterns of electromagnetic radiant energy and other phenomena" [69].

Data is acquired through an optical sensor, most typically cameras that have CMOS or CCD detectors. The amount of data obtained in an image is limited by the number of pixels and bits in the analog-to-digital (A/D) converter. The pixel is the smallest spatial unit of an image. The value of a pixel is defined by the A/D resolution and the mapped gray shade of the particular image location [44].

Figure 2 shows a flowchart of the coordinate system transformation from world coordinates to the image plane when a photograph is taken. The extrinsic camera parameters transform the camera coordinate system using the rotation and translation matrices. Mapping to the image coordinate frame is done using the intrinsic camera parameters, which depend on the lens focal length, the camera principle point, and radial and tangential distortions [99].

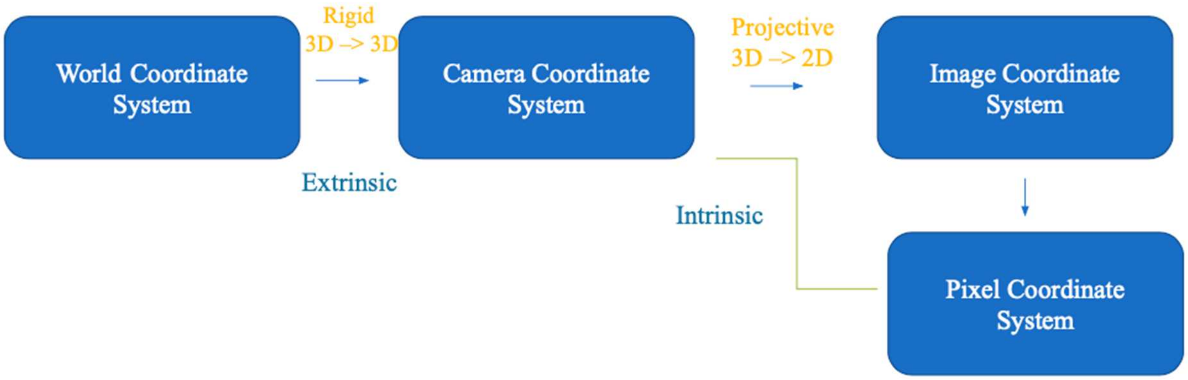


Figure 2. Photogrammetry process flow chart.

The digital photography age began at the end of the 20th century. Charged couple device (CCD) sensors, a noble prize invention, were the first to be used as the digital platform for digital cameras. With the creation of smartphone cameras, complementary metal oxide semiconductor (CMOS) detector arrays, invented by Olympus [68], have become more widespread. Today's iPhones utilize these sensors as detectors, with pixel resolution reaching as low as 0.7-0.8 µm [67].

Monochrome (greyscale) color sensors store information in 256 levels or more, depending on the camera's resolution. The image obtained from these sensors is grayscale. On the other hand, color sensors need three values, the red, green, and blue (RGB) coordinates. These three are also represented in 256 levels, 1024, or 4096, depending on the resolution of the camera sensor. Color cameras have three channels, red color, for example, is described by the vector $[255, 0, 0]$ on the pixel, and white is represented by $[255,255,255]$ [100]. Apple's iPhone uses RGB color sensors with a prism design, where light is split into three separate sensors for each pixel to capture the red, green, and blue components [101].

We have to perform a geometric reconstruction to reconstruct the 3D object space of our assembly from the images. The accuracy and efficiency are determined by the similarity of the image bundle with the principle rays that entered the lens when the photograph was taken. This correspondence is calculated and improved using camera calibration. Photogrammetric applications include orthophotos, where photogrammetry calculates the distance of the features in an image [102]. Other applications include topographic maps, where the shape of the land surface is reconstructed using photogrammetry, and architectural 3D reconstruction, where the architectural models and ancient buildings are preserved using 3D data that can then be modeled and used for virtual tours [44].

Figure 3 shows that photogrammetry, when treated as a black box, delivers a point cloud or mesh of the reconstructed 3D scene from input images. The basic principle behind photogrammetric measurements is triangulation, whereby light rays from the object are triangulated to reconstruct the 3-D points in real space geometrically. In principle, at least two images of the same point are needed to calculate the intersection of the rays into a 3-D coordinate [103]. This is the basic principle; however, bundle adjustment and camera calibration are also needed to calculate camera angles and characteristics to refine the 3D locations of the reconstructed scene [104].

Input images
Photogrammetry
3D world coordinates, point cloud / mesh

Figure 1. Photogrammetry input and output.

2-1 PHOTOGRAMMETRY COMPARISON WITH OTHER TECHNIQUES

The laser scanner works like photogrammetry in that triangulation is performed to gather 3D data. A collimated light beam is projected onto the target, the beam is in the form of a point or a line, and a CCD imaging device then detects the object's profile shape. As the laser sheet over the object, 3D data is collected [109]. Laser scanners offer higher accuracy over large spaces compared to photogrammetry. However, the data can be very fuzzy over highly textured surfaces.

Several studies have been done comparing laser scanners with photogrammetry. One study compared terrestrial laser scanners with data captured for photogrammetry using a 20-megapixel camera [4]. The most promising advantage of photogrammetry is easy setup, reduced data acquisition times, and reduced equipment expense compared to laser scanning approaches. The cost of laser-based systems is high, and the number or availability of such devices is limited. The scanners offer greater measurement resolutions and wider angle of coverage. The resolution here is the distance between the scanner’s measuring points.

Structured light measurement methods are active 3D imaging methods that actively illuminate the object to aid 3D reconstruction. A device is used to project structured light onto a surface. Various 3D scanning technologies have been developed; each scanning strategy comes with its pros and cons. The properties and characteristics of the object to be measured determine which technique to be used. One of these techniques is light modulation, where modulated light is projected onto a surface in a sinusoidal pattern. The reflected light is detected by a sensor [105]. Another example is binary patterns and gray coding, where black and white stripes are projected onto the surface [108]. Chapter 1 briefly overviews the different structured light techniques and their measurement accuracies.

Structured light techniques have a smaller measurement volume; while it is possible to use them in larger areas, that comes at a computational cost and data storage problems [106]. Furthermore, the equipment needed to perform structured light measurements is costly and requires specialized labor. The scene also needs to stay static for a period when light is projected on it.

On the other hand, image-based quality inspection systems are among the most popular quality control systems since they can be easily implemented at a lower cost. Especially for small companies and startups, buying high-quality, industrial-grade cameras is relatively inexpensive. The cameras can also be controlled remotely, thereby decreasing on-site personal time. Photogrammetry also requires a minimal workforce in that cameras can be left at the scene at different locations, and image acquisition can be automated. There is no necessity for skilled labor to take images from separate areas.

Analysis of 3D image-based data acquisition methods, including laser scanners, photogrammetry, and structured light systems in an industrial environment, shows that when selecting a particular method, the conditions of the work environment, i.e., changing illumination levels and moving geometrical surfaces of interest need to be considered. Methods of active illumination, like structured light techniques are preferable because inspection systems become less sensitive to changes in the illumination conditions of the industrial environment [5]. Although structured light methods are preferable as they also lead to lower 3D point cloud uncertainty, they require a skilled worker and expensive equipment and software to work. Certain factory conditions offer low flexibility on the availability of resources, such as computing devices and electricity, whereby offline data acquisition needs to occur, and photogrammetry is a good fit.

Reasonable lighting conditions on a factory floor, combined with the availability of high-processing graphics processing units (GPUs), make the photogrammetry-based inspection a good fit. Close-range photogrammetry in industry is a relatively new concept providing highly accurate 3-D measurements that became popular in the 1980s. For large objects greater than 10 m, photogrammetry offers a precision of 1:500,000 [6].

Furthermore, while contact measurement methods like the CMM and non-contact methods like laser scanners offer low measurement uncertainty, they come at a hefty price. A study on geometrical measurements in car testing laboratories studied the suitability of laser scanners and photogrammetry techniques to perform these measurements [8]. The study points out that the geometrical parameters of the cars in these laboratories are set by ISO 612:1978 and the dimensional tolerances mentioned are high enough to compensate for the relative lower uncertainty offered by photogrammetry and laser scanners.

2-1-1 COMMERCIAL PHOTOGRAMMETRY PRODUCTS

Geodetic (Melbourne, FL, USA) offers commercial photogrammetry software and devices such as V-STARS to measure aircraft engines, boxes, and large ship parts, among others [9]. The package includes specialized cameras, scale bars, and coded targets. Figure 4 [60] shows the package, which provides photogrammetry targets, handheld camera devices, and computer software.

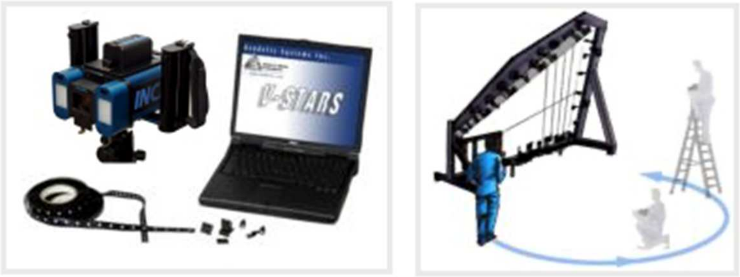


Figure 4: V-STARS photogrammetry device

Pix4D [61] and DroneDeploy [62] software is used for unmanned aerial vehicle UAV land surveying by using drones. Pix4D offers a range of products, from mobile photogrammetry apps to data capture hardware, such as a camera mounted on a crane, which can be controlled remotely. Similarly, DroneDeploy offers data acquisition with autonomous drones and programmed robots to walk through the site and capture data with photogrammetry data acquisition and analysis software.

For large-scale close-range photogrammetry, Bentley ContextCapture [63] offers products for the civil industry sector. Their systems create 3D reality meshes or point clouds from input images to be shared and analyzed over the cloud in the civil engineering industry. CapturingReality [64] is another mobile photogrammetry app for close-range photogrammetry applications. Their software and apps cost hundreds to thousands of monthly subscription costs.

Thus, there is an opportunity in the market for an economical alternative. Cell phone computational abilities allow for sophisticated applications and can be readily accessible, as virtually everyone in the 21st century has access to a smartphone and smartphone camera. The combination provides on-spot 3D reconstruction; however current commercial apps require a hefty monthly subscription. Thus, we focus on exploring free resources to create a pipeline that can be interfaced with the whole inspection model, including data processing, comparison with the reference, and visually displaying the points for the user.

2-2 USE OF OPEN-SOURCE LIBRARIES

Two data analysis routes for performing photogrammetry are commercial software and open-source alternatives. From a consumer point of view, commercial and freeware software solutions are black boxes, and it is impossible to customize and modify the software for specific needs. Whereas, open-source solutions provide complete access, we can modify multiple view geometry algorithms, and they can be interfaced into an over-arching software architecture [49]. Furthermore, open-source libraries offer complete transparency and promote research, as anyone can test and improve the algorithms, thus developing a supportive community for a variety of applications of the library.

Keeping this in mind and as mentioned in the introduction, our focus is on making the photogrammetry pipeline economical, which is why we pursued the open-source libraries OpenMVG [18] and OpenMVS [23]. Both open-source libraries were combined to obtain 3-D dense point clouds from images. Photogrammetry can be divided into two main categories. One method reconstructs 3-D point clouds with prior camera position and angle knowledge. The second category uses a structure from motion (SfM) pipeline, which simultaneously calculates the camera pose and the 3-D shape using feature detection in images, after which feature matching occurs. OpenMVG offers a collection of integrated small libraries that can be used in a pipeline to perform 3D reconstruction from images using SfM [49]. Similarly, OpenMVS is integrated with this pipeline to obtain a dense point cloud.

The OpenMVS library is based on the work of Moulon Pierre, Monasse Pascal, and Marlet Renaud [23]. The library is based on the structure from motion (SfM) algorithms, which take, as input, images to yield reconstructed points in the 3D space. Instead of taking in images incrementally and building the 3D space using a globally fixed threshold to estimate the object (3D points in space), the library uses a '*contrario*' method that adapts thresholds for each input image and model estimation [46].

One of the first steps while computing a 3D structure from images is to follow the position of a characteristic point in a set of images. To find the correspondences of these points in multiple images is called tracks. The library uses the union-find [111] algorithm to fuse two-image correspondences into multi-view tracks [47].

The multi-view SfM pipeline used in the library uses a modified global calibration approach. Conventionally used incremental methods for estimating the orientations of the images are subject to drift because errors are accumulated [48]. The modified global calibration approach is based on the fusion of the relative motions of the image pairs. The camera orientations are calculated using a trifocal tensor estimation method, and the camera positions are calculated through a translation registration method [48].

The OpenMVG library is written in C++ and uses the CMake build system to create executables for windows and advanced RISC machine (ARM) targets. OpenMVG utilizes the Eigen library [51] to perform linear algebra and the ceres-solver [50] to do the bundle adjustment, a form of large-scale non-linear minimization. Being open source and offering extensive documentation, it is easy to use the library with other software and custom applications [49].

OpenMVG recovers camera poses and reconstructs a sparse 3-D point cloud from input images. We interface this output with another open-source library, OpenMVS, which has algorithms to recover the entire surface of the scene to be measured. This part of the photogrammetry pipeline uses the estimated camera poses and the sparse point cloud outputs from the OpenMVG step and establishes an estimate of a dense 3D point cloud of the object.

2-3 STRUCTURE FROM MOTION PIPELINE

We follow the SfM pipeline to perform photogrammetry. The pipeline starts with feature extraction, where feature identification algorithms are used to identify features in the images. A 2D point location can describe each region in the image, its scale (the zoom scale of the image), and its orientation. We can detect blob regions, scale-invariant points, corner regions, or affine invariant regions. The algorithms SIFT [53], AKAZE [54], FAST keypoints [55], and tree-based morse regions [56] are a few examples of the feature extraction and description algorithms that were explored. Once these features are identified in images, they are matched and identified in the set of input images. This image matching is done using brute force methods [112] or ANN-kD trees [52]. The algorithms mentioned are used to compute the nearest 3D points in the image pairs or to find the corresponding points on the images by matching the features across a series of images [49].

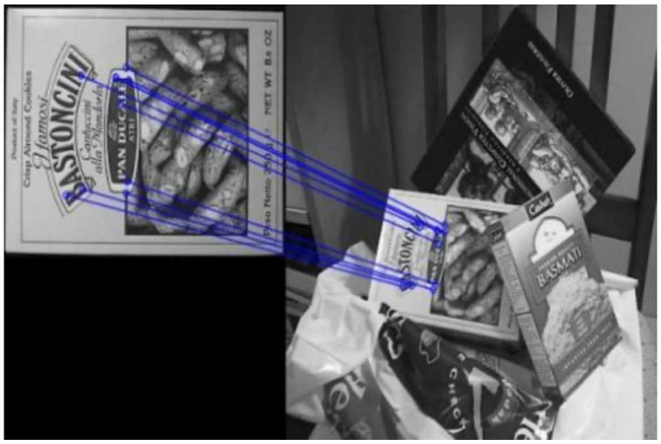


Figure 5: Feature matching in image processing [113].

Figure 5 [113] shows two images from the OpenCV library going through feature matching. Some multiple view geometric constraints such as relative pose, absolute pose, and triangulation are also checked to filter out the set of matching feature points between the image pairs. Figures 6 (a) and 6 (b) show the SfM pipeline in more detail.

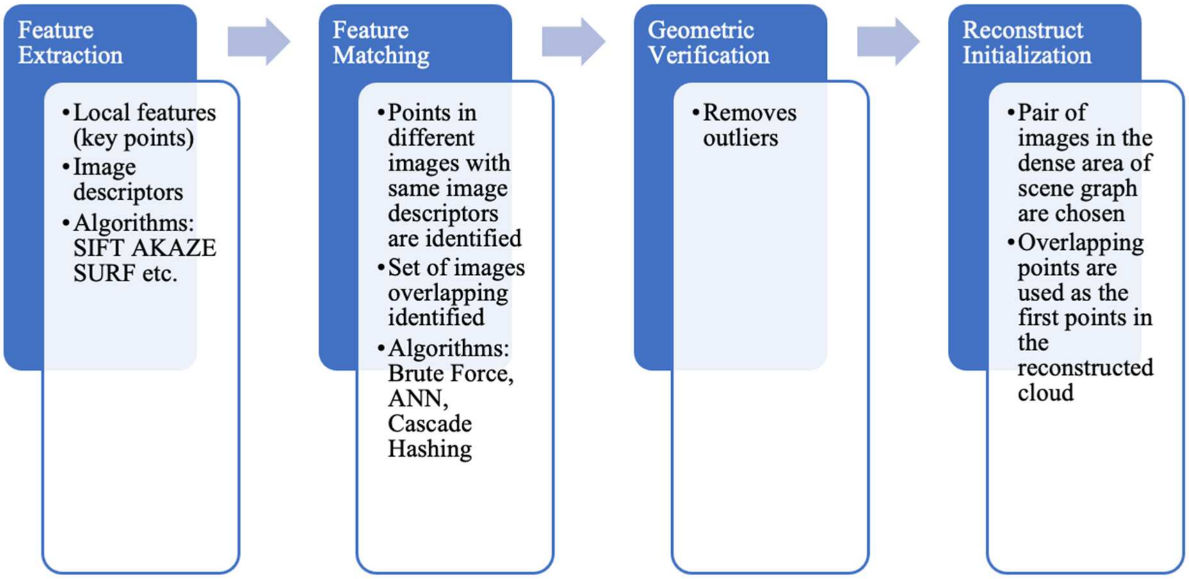

Feature Extraction
•Local features (key points)
•Image descriptors
•Algorithms: SIFT AKAZE SURF etc.
Feature Matching
•Points in different images with same image descriptors are identified
•Set of images overlapping identified
•Algorithms: Brute Force, ANN, Cascade Hashing
Geometric Verification
•Removes outliers
Reconstruct Initialization
•Pair of images in the dense area of scene graph are chosen
•Overlapping points are used as the first points in the reconstructed cloud

Figure 6 (a) Structure from motion steps for 3D reconstruction (part 1).

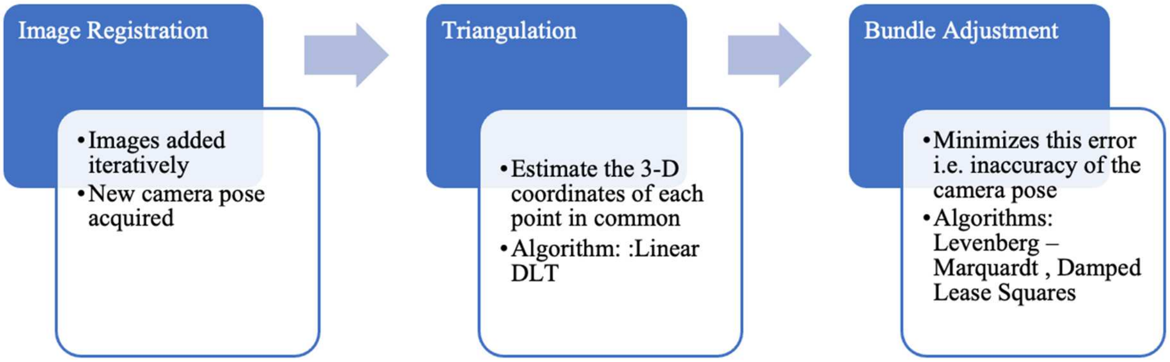


Image Registration
•Images added iteratively
•New camera pose acquired
Triangulation
•Estimate the 3-D coordinates of each point in common
•Algorithm: :Linear DLT
Bundle Adjustment
•Minimizes this error i.e. inaccuracy of the camera pose
•Algorithms: Levenberg – Marquardt , Damped Lease Squares

Figure 6 (b): Structure from motion steps for 3D reconstruction (part 2).

2-3-1 FEATURE DETECTION ALGORITHMS

As mentioned in the section above, the features in the images need to be identified and described. We use the scale-invariant feature transform method (SIFT) [57], introduced by D.Lowe and now included in the OpenCV library. The feature detection algorithm is robust to occlusions, as neural networks can be used to predict data [114], and it is highly efficient.

A scale space is produced from the image by the convolution of a gaussian kernel, which is used at different scales; this is used to create several octaves, as shown in Figure 7 [58].

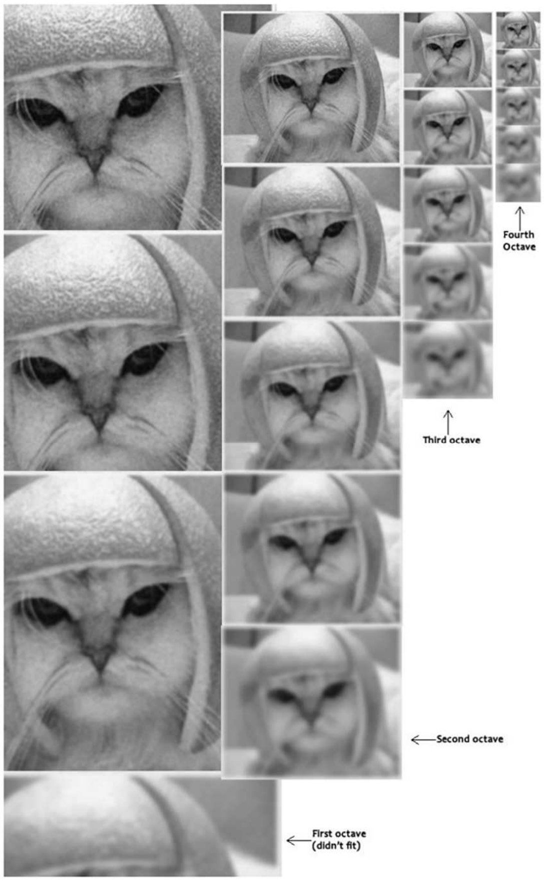


Figure 7: Concept of octaves in the images illustrated [58].

The image $I(x, y)$ is first blurred using a gaussian blur operator $G(x, y, \sigma)$, where x, y are the coordinates and σ is the scale, which controls the amount of blurring. The blurred image $L(x, y, \sigma)$, can be written as

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y)$$

$$where, G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{x^2+y^2/2\sigma^2} .$$

These blurred images are then used to generate a difference of Gaussians. The key points are created by comparing each pixel with its neighbors in the same, next, and previous scales. If it is a local maximum, then it is considered a possible key point [58]. These key points are then filtered and assigned an orientation, location, and scale.

2-4 DENSE POINT CLOUD RECONSTRUCTION

The SfM pipeline gives a sparse point cloud which is then used for dense reconstruction. As mentioned earlier, OpenMVS is used in a pipeline following OpenMVG to perform this dense reconstruction. This library targets the last part of the photogrammetry pipeline by reconstructing the detailed surface of the scene or object to be reconstructed. The library is based on the depth-map merging method proposed by Shuhan Shen [66]. This algorithm is suitable for large-scale reconstruction compared to other methods like surface evolution methods and feature point growing methods.

The depth map merging method consists of a few steps. The first step is the stereo pair selection, whereby the images are sorted based on the angle between the principal view directions of any two camera locations and the distance from the cameras' optical centers.

The second step is to compute a depth map, i.e., every pixel in an image is assigned a depth (z-coordinate) and a patch normal. For each pixel 'p', its neighboring pixel 'q' is used to calculate the corresponding pixel in the reference image [115]. A matching cost is then calculated. This depth map goes under a refinement step, after which the maps are merged and go under some checks for the neighboring pixels to create a dense point cloud [66].

2-5 PHOTOGRAMMETRY PROCEDURE TEST RUNS

2-5-1 TEST ARTIFACT

To test the proposed pipeline, we designed and had a fabricated steel-based assembled system of welded components onto a steel plate. The method we described using photogrammetry can be applied to various assembly applications involving different materials and scales. A CAD model of the artifact used here was made in Creo parametric [15], and a real-life model was constructed, whereby the individual features were welded onto a base steel plate.

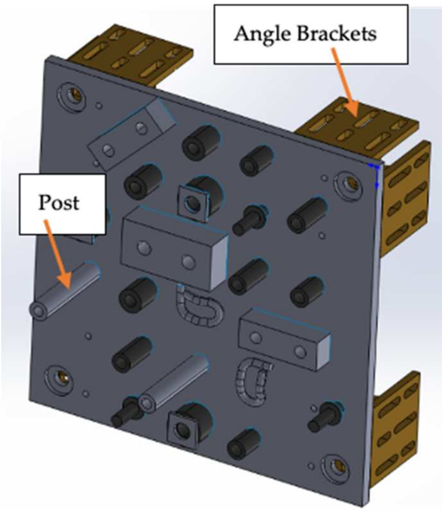


Figure 8. CAD model of the artifact to resemble assembly line features.

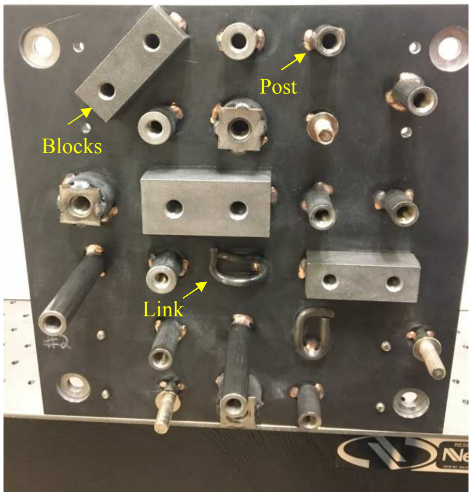


Figure 9. Plate constructed using the CAD design to be used to demonstrate the pipeline.

Figure 8 shows the CAD model of the artifact. The system is 30 mm by 30 mm and contains assembled blocks and posts that serve as the assembled components of interest with heights as large as 40 mm. Angle brackets are also shown in the CAD to facilitate mounting onto an optical bench. Figure 9 shows the steel plate constructed from the CAD model where blocks, links, bosses, posts, etc., were welded onto the plate at specific locations. It is bolted to an optical bench, representing a fixed location in an assembly line. The plate is made of steel and has a rough surface texture, representing a steel assembly in various automotive and aerospace applications. The system was designed to have standard components welded onto a steel manifold that would be consistent with typical automotive/aerospace assembly applications.

2-5-2 SPARSE POINT CLOUD RECONSTRUCTION

For data acquisition, in keeping with our goal of an inexpensive and accessible system, we used a smartphone camera, as described earlier in Section 2.1, to collect a sequence of images. The camera is used with a fixed focal length by turning off auto focus and filters in the smartphone camera are disabled. The focal length is fixed, and 60 images of the artifact are taken from different angles with an approximate shooting distance of 1 m. Multiple images with around 90 degrees spread and about 50 % image overlap were taken to improve the reconstruction accuracy [16]. Exact knowledge of the camera positions for each image is not needed because this is determined through bundle adjustment in the photogrammetry process (the camera extrinsic parameters, i.e., rotation and translation of the image plane away from the camera). For intrinsic camera parameters, the photogrammetry pipeline from OpenMVG self-calibrates the camera, using an estimated focal length mentioned in Equation (1). Normal indoor room lightning conditions were used.

Since we don't know the exact intrinsic camera parameters, some estimates were provided to the pipeline. The following Equation [17] was used to estimate the focal length in pixels

$$focal_{pix} = \frac{\max\left(w_{pix}, h_{pix}\right) \times focal_{mm}}{ccdw_{mm}}, \tag{1}$$

where $focal_{pix}$ is the camera focal length in pixels, $focal_{mm}$ is the focal length in mm, $w_{pix}\ and\ h_{pix}$ is the image's width and height in pixels, and $ccdw_{mm}$ is the camera sensor width size in mm. Keeping the focal ratio, i.e., $(focal\_mm)/(ccdw\_mm\ )$ equal to 1.2 gives a reasonable starting estimate by setting the FOV of the camera equivalent to 45 degrees. This estimate is just a starting point for the self-calibration step part of the bundle adjustment.

After the images are taken, they are used for the first step in the SfM pipeline, i.e., feature extraction. In this step, we used SIFT, which locates key points of features in the images and assigns a descriptor to them. The algorithm is scale-invariant, according to Lowe [20]. The scale invariance is achieved using various octave or scale levels, convolved with a blurring function. The difference of Gaussians (DOG) is then calculated to extract prominent key points in the images [20]. Once the locations of the key points are calculated, their orientations are also estimated to make it rotation invariant. Gradient magnitudes and directions are calculated using the neighboring pixels around the key points. Once the key point locations and orientations are established, feature matching occurs. Features extracted using SIFT have shown to be robust with images with noise, changes in illumination, and across a range of affine distortion [20].

The fast cascade hashing method is used for feature matching across images as the next step in the SfM pipeline. This method is an approximate nearest neighbor (ANN) method based on hashing, which uses a three-step process to map an image into its binary code [21]. Once image matching is done, a global bundle adjustment method [22] is utilized, and a sparse point cloud is obtained.

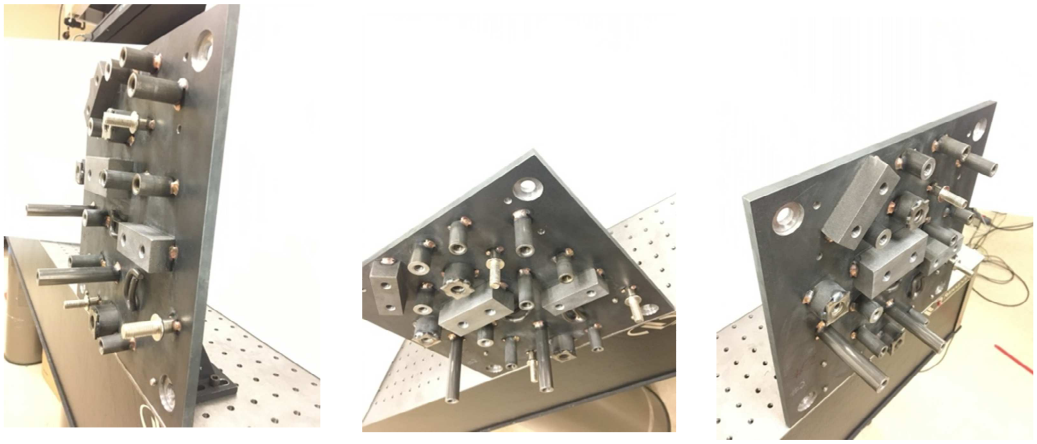


Figure 10 Images of the artifact taken at different camera angles.

The sparse point cloud obtained using an incremental SfM pipeline is shown in Figure 12. The yellow dots in the figure show the camera positions around the artifact where the images were captured. The images must have overlapping regions to triangulate the same point in at least two images for 3D reconstruction of the point. Figure 11 shows the calculated camera poses.

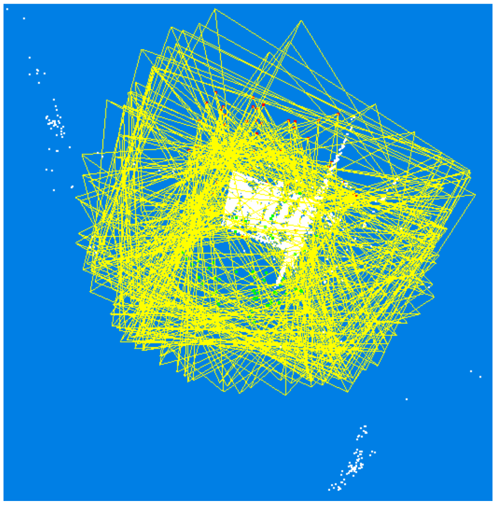


Figure 11. Camera poses are calculated and shown.

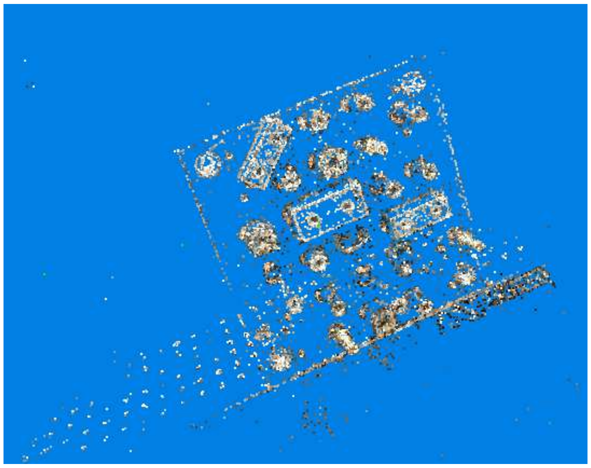
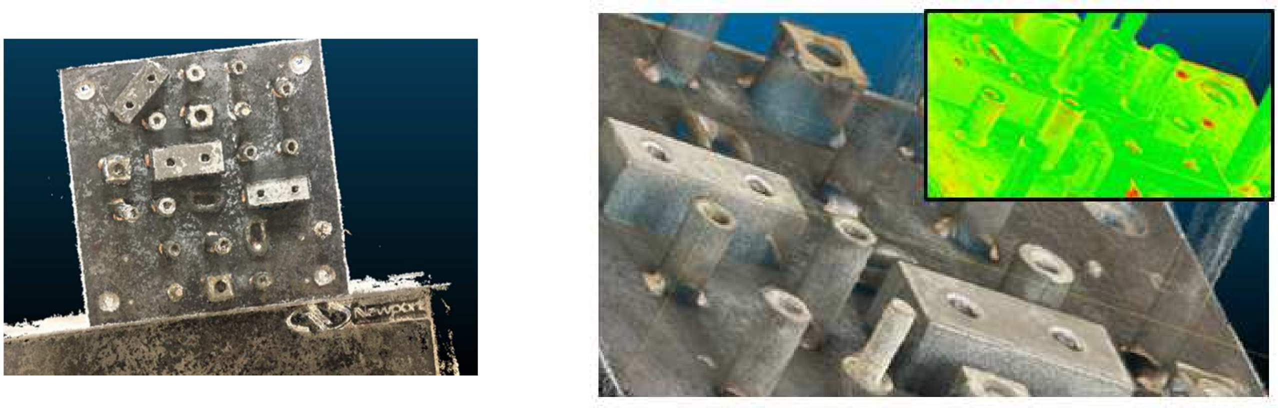


Figure 12. Sparse point cloud generated using photogrammetry.

2-5-3 DENSE POINT CLOUD RECONSTRUCTION

To obtain a dense point cloud, we use another library called OpenMVS [23]. A depth map merging method was performed here. The method uses data from SfM and the previously constructed sparse point cloud to create depth maps and refine them. These depth maps are refined using PatchMap, filtered, and merged to form a dense point cloud. PatchMap performs nearest neighbor matches between different image regions [116].

This method works well on large-scale objects [30]. Figure 13a shows the resulting dense point cloud. Figure 13b shows the point cloud with a distance map relative to the CAD model shown in the insert. The distance between the nearest point on the point cloud and the CAD model lie within −0.75 mm to 0.75 mm [31]. As seen in both figures, the sparse point cloud acts as a skeleton for dense reconstruction to fill in the points on the surface. The dense point cloud shown is reconstructed with surface texture details and the RGB values of the points visible.



13 (a)

13 (b)

Figure 13. (a) Dense point cloud, (b) Close-up of the point cloud with distance map in the insert.

For assembly position error estimation, the geometric shape of the components (the boss, block, and link on our plate) must be well constructed. We want to calculate the component's position relative to a global coordinate system for comparison with the reference, which is the nominal location. As shown in the figures, the components on the plate are visually well reconstructed in the 3D point cloud. The individual component reconstruction however is not apparent here, we will see the effects of noise on the individual points and the deviations from CAD in the next two chapters.

Figure 14 shows the distance map in more detail. A distance map was calculated using the photogrammetry data and its CAD model to obtain an idea of the noise and deviation of the data from its reference. The map shows a distribution of difference values with an approximately gaussian shape with a full-width half max (FWHM) of $0.6\ mm$. The deviation in the map of the fabricated artifact can be attributed to noise from the photogrammetry process and deviation of the actual component shape compared to its reference CAD. As mentioned above, the distance map has a deviation of FWHM of $0.6\ mm$.

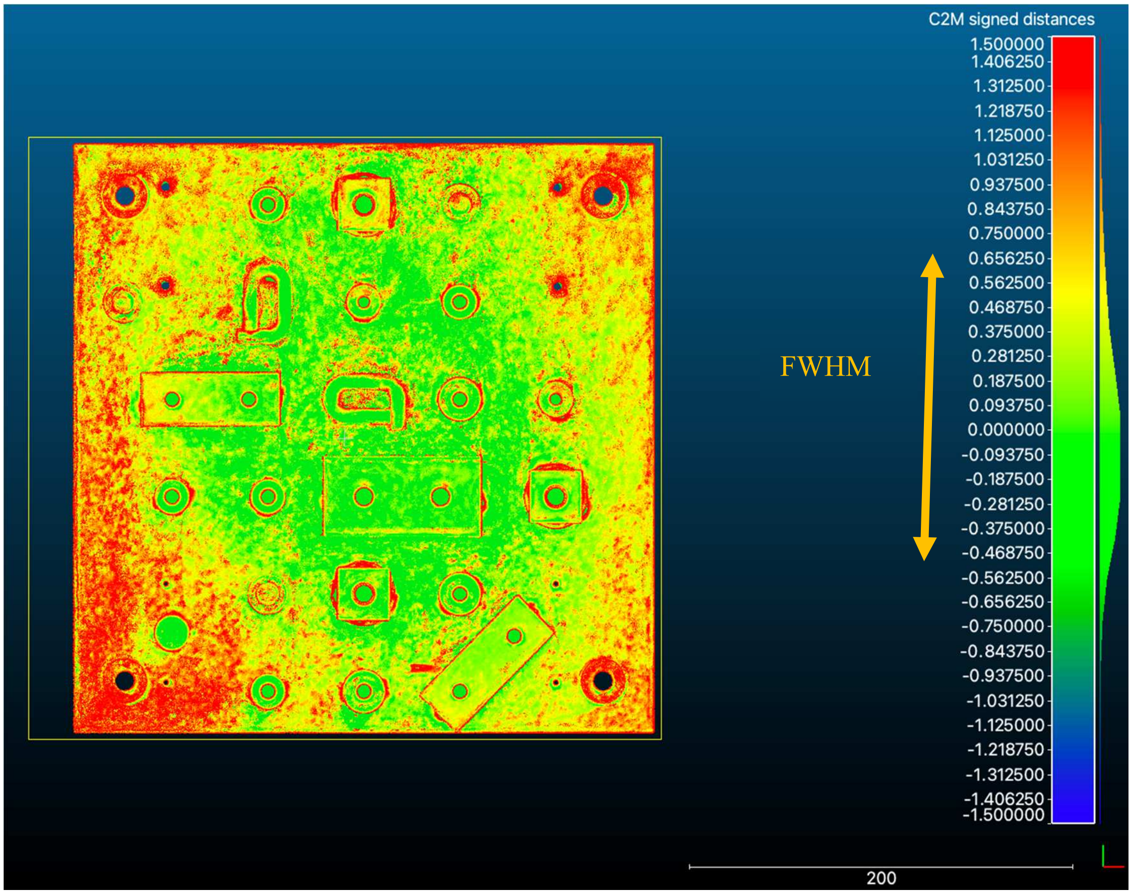


Figure 14. Distance map to calculate the deviation of the data from reference.

The above method verification successfully showcased an economical and flexible method, using just a smartphone camera and a laptop for 3-D measurement of an assembled part to be inspected in an industrial assembly line application. The data acquisition was straightforward and did not require the cumbersome and time-consuming step of adding coded targets to the object. The dense 3D point cloud reconstruction process took approximately 10 minutes to generate from images.

The next chapter focuses on processing and comparing this 3D measurement data with a CAD model as a reference to estimate the position error o bnf assembled components on the part.

CHAPTER 3 POSITION ERROR MEASUREMENT USING CAD AS REFERENCE

3-1 INTRODUCTION

With the revolution of industry to I4.0, there has been a shift from less variety of products but more quantity to more variation, customized products, and low quantity [70]. Industry I4.0 refers to smart factories where product manufacturing and distribution have improved with the integration of machine learning, cloud computing, and internet use [117]. This advancement has affected the quality inspection industry and has led to the adoption of Assembly System 4.0 [70]. More customizations make it harder to manage production, and more agile and flexible quality inspection systems are needed.

Quality inspection can be divided into three parts, pre-production inspection, in-line inspection, and final inspection. Pre-production inspection focuses on testing raw materials, such as the material weight, resistance, and dimensional stability, before they enter the production phase. The in-line inspection ensures that the components are accurately processed or cut after each critical production step. Final inspection happens at the end of the production process before the part is handed over to the buyer or consumer [71]. Since we focus on assembly line errors, our work targets final inspection or in-line inspection for an intermediate assembled step.

Although quality inspection encompasses an extensive array of tests and checks, this dissertation focuses on the quantitative inspection of a components' position error in the production assembly stage. Assembly error is the variation of the installed position of a component compared to the reference position required by the design specification.

When a component is assembled onto the parent system, all 6 degrees of freedom (DOF) of rigid body motion are involved. Here we reduce the general 6 DOFs and define the position error measurand as the displacement of the center of mass of the component on the assembly from the nominal position defined in the reference CAD model. The pipeline presented here delivers a quantitative estimate of all 6 DOFs of the component; thus, the work could easily be extended to define other related measurands. Unique position error estimates of components on the assembly requires that the measured point cloud of the assembly be aligned with the CAD model in some way. This is done by defining a global coordinate system in the CAD model, realizing an estimate of the same origin and orientation in the measured point cloud data, and then transforming the measured point cloud so the coordinate systems overlap. This establishes unique positions for the assembled components in the global coordinate system, thereby allowing for unique estimates of position errors. This process is discussed later in the chapter.

The position error measurand in our application is found through registering together two-point clouds of the individual components. This registration results in a transformation matrix, which includes the moving point cloud's translation and rotation to align with the reference point cloud. The distance of the center of masses between the two-point clouds will serve as our position error measurand.

While coming up with a solution for the position error inspection problem, we have made the process independent of component shape and utilized the availability of CAD models or any reference point cloud for comparison purposes. Furthermore, with our goal of making the pipeline cost-effective, flexible, and adaptable, we relied on economic and customizable codes. Commercial software was also explored briefly while performing this research.

Point cloud data needs to be visualized and processed for metrology applications to pass quality inspection. Numerous platforms have been developed for point cloud visualization and processing. They include commercial software packages designed by manufacturers of point cloud measuring devices and independent commercial software packages. GOM Inspect, FARO scene, Polyworks, and PointCab are a few such examples [5]. These software packages offer advanced point cloud processing and error measurement capabilities, but they come at a hefty price.

Apart from commercial software, two well-known open-source libraries are available for point cloud processing, PCL (point cloud library) and Cloud Compare. MATLAB, though not open-source, also offers an economical and open-ware coding platform.

This chapter focuses on finding and quantifying component position errors while performing a quality inspection in the industrial assembly line. We are targeting millimeter-scale errors in position with submillimeter uncertainty. We begin the discussion with our study of and approach to point cloud registration.

3-2 POINT CLOUD REGISTRATION THEORY

Remote sensing gathers data in the form of 3D point clouds. Typically, multiple sensors are used to acquire the data to reduce data acquisition time and increase accuracy [72]. The alignment needed to integrate the numerous data sets is called registration. Another application for registration is when two independent but corresponding (source and target) point clouds need to be aligned, as when making a comparison for quantitative inspection and the model is aligned with the data for comparison. Registration is an essential step in estimating the actual component location compared to the nominal location. [72].

Research in the field of 3-D registration began in 1970, initially in the military, and now the applications range from medicine to computer vision [77]. Registration can be divided into coarse and fine registration. Coarse registration ensures that the two sets of point cloud at least overlap and that the source point cloud is oriented somewhat in the same direction as the target data set. In comparison, fine registration optimizes the overlap of the two-point clouds through iterative techniques or random sample consensus methods [77]. Fine registration algorithms rely on good initial alignment to avoid getting stuck in local minima.

In our application, the coarse and fine registration terms are slightly intermixed since we are using fine registration algorithms in both steps. In this dissertation the coarse registration term will be used for coordinate system alignment and fine registration term will be used for registration done to realize the individual component position estimates and errors.

Before fine registration can take place, coarse registration is used to achieve a good initial guess. Since we will be using ICP on the component itself, not the background, we'll need to make sure our coarse registration has enough overlap, so it does not include the neighboring elements when masking out the component of interest.

3-2-1 COARSE REGISTRATION

Generally, the first data analysis step is a coarse registration between the measured point cloud data and the reference CAD model. In photogrammetry, coarse registration is usually achieved with three coded targets in the scene that define an origin, an x-axis, and an *x-y* plane to establish the coordinate system (see Figure 15) [24]. Depending on the application, many algorithms and strategies are used for coarse registration. In our measurement, the method relies on the shape of the parent substrate on which the components are assembled. Our demonstration artifact is a plate on which posts, blocks, links, etc., are attached. We pursued plane detection strategies as the first step of our coarse registration because general background shapes typically have approximately local flat regions. Rather than apply coded targets, we also use specific features on the part itself (e.g., two holes) to complete the information needed to define the data global coordinate system.

We used a combination of plane detection and ICP fine registration to perform this data global coordinate system alignment which we are referring to as coarse registration for our application.

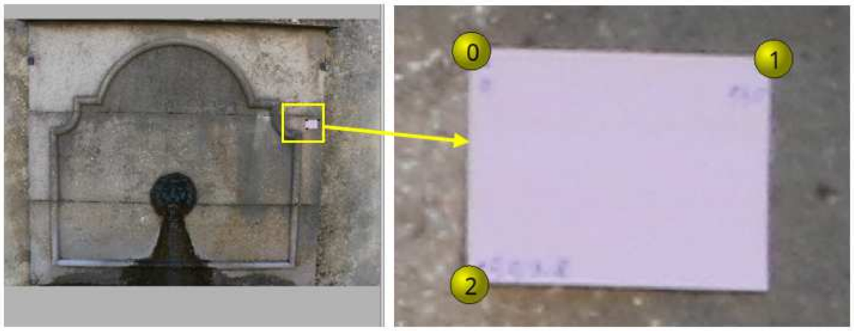


Figure 15. Coordinate system alignment using targets placed in the scene where corners of the target are used to lock the origin and the x, y, and z-axes.

3-2-2 FINE REGISTRATION WITH ITERATIVE CLOSEST POINT THEORY

3-D registration comprises pose estimation, alignment, and motion estimation. Most of the literature focuses on certain types of shapes, such as polyhedral models, quadratic models, and point sets with known correspondences [70].

Gilbert and Foo [74] did work on computing the distance between two objects, which could have typically been used in shape-matching techniques. However, these methods rely on decomposing the shapes into sub-bodies which is not a simple task to perform on CAD models [70].

The iterative closest point algorithm is a free-form surface matching method that can be generalized for *n* dimensions. It does not require any extracted features or surface normal information [73]. This shape registration method can be applied to 3-D point clouds, line segments, and curves, both parametric and implicit [73].

ICP finds an optimal rigid transformation to align source points with target points for rigid body registration between two sets of points. This method's weaknesses are its sensitivity to outliers and slow convergence. Much work has been done to improve ICP drawbacks, such as optimizing the convergence speed by treating the point-to-point ICP registration as a majorization-minimization algorithm [75]. Noise effects are also discarded by eliminating erroneous correspondences between the points in the data sets by using learning-based methods based on calculating the point distances and the angles between the normal [76].

Zhang and Yao [76], give a brief overview of the traditional ICP method. The classical ICP algorithm works as follows, given two sets of point clouds containing the points $P = \{p_1, p_2, \ldots\ldots, p_m\}$ and $Q = \{q_1, q_2, \ldots\ldots, q_m\}$, we treat P as the source point cloud and Q as the target data set. A rigid transformation of P is calculated using rotation, R, and translation T matrices. The distance between the transformed point $Rp_i + t$ to q is calculated, and the function is minimized.

This minimization function is then solved using an iterative method. The first step to solving this minimization is to find the closest point, $q_i^{(k)}$ in the target data set, this point can be written as

$$q_i^{(k)} = argmin\left|R^{(k)}p_i + t^k - q\right|$$

The second step is to update the transformation by minimizing the function below:

$$(R^{k+1}, t^{k+1}) \ = \ argmin\left|Rp_i + t - q_i^k\right|^2$$

To solve the above function, singular value decomposition (SVD) is used. This is done iteratively until the mean square change falls below a particular threshold value. This threshold controls the target precision of the registration process [73].

This classical ICP is 'point to point', as the alignment minimizes the distance between the source points and target points. Another way to use ICP in 3-D space is to compute the 'point-to-plane' distance minimization. The 'point-to-plane' approach minimizes the distance between the points in the source to the tangent planes in the target data set.

Given any rotation and translation, the ICP algorithm converges to the local minimum, which might not always be the desired global minimum. To solve this global minimization problem, people perform coarse registration. This step can be done by iteratively performing ICP on the source point cloud or by doing transformations so that the center of masses of both point clouds align [73].

Some of the significant advantages of ICP algorithms are that it can handle all 6 degrees of freedom, are independent of shape variations, and can handle noise levels of standard deviations up to 10 percent of the object dimensions [73].

3-2-3 POINT CLOUD REGISTRATION METHODOLOGY

Positioning errors are calculated by comparing the 3-D point cloud of the object with its CAD model. This involves processing the 3-D data by removing background features, implementing coordinate system alignment to a global coordinate system, and finding deviations with respect to the CAD model.

3-2-3-1 GLOBAL COORDINATE SYSTEM ALIGNMENT

Before the position errors of assembled components can be estimated, a global coordinate system must be defined. As discussed above, the CAD model and the 3D photogrammetry data need to be registered/aligned (axes and origin). This coarse alignment provides an aligned starting point from which the position error of features over the assembly can be evaluated. This step starts with using the plane of the steel plate to lock two degrees of freedom of the coordinate systems. The plane is extracted from the 3D point cloud data set using a least-squares best fit to the plane equation

$$ax + by + cz = d, \tag{2}$$

where the plane normal is $n = (a, b, c)$ and d is the distance from the origin to the plane.

The MATLAB function 'pcfitplane' [120] is used to fit a plane to our photogrammetry point cloud data. The reference vector of [0, 0, 1] is given as input to extract the horizontal point clouds on the plane. Figure 16 shows the extracted point clouds where the red surface is used to define the plane parameters of these points. Figure 17 shows the same plane surface on the whole point cloud. The plane parameters are then used to find the rotation such that the normal to the plane aligns with the z direction, and the plane becomes parallel to the x-y plane. The value of z is zero at this height and defines the x-y plane.

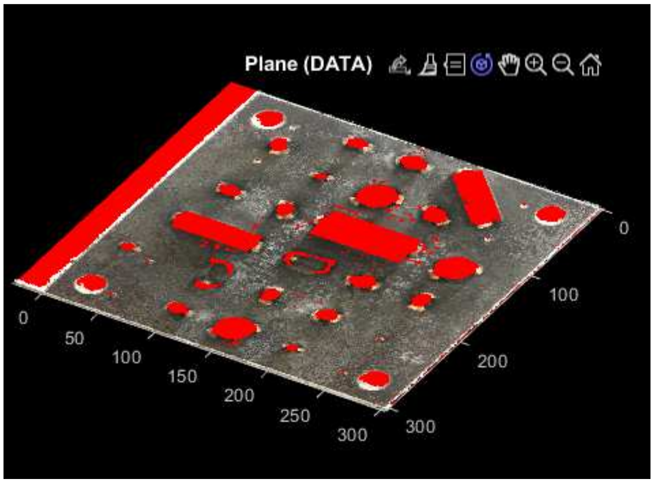


Figure 16. Points on the x-y plane are shown in red.

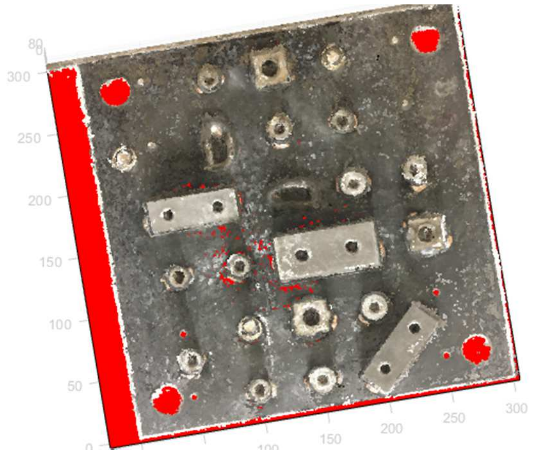
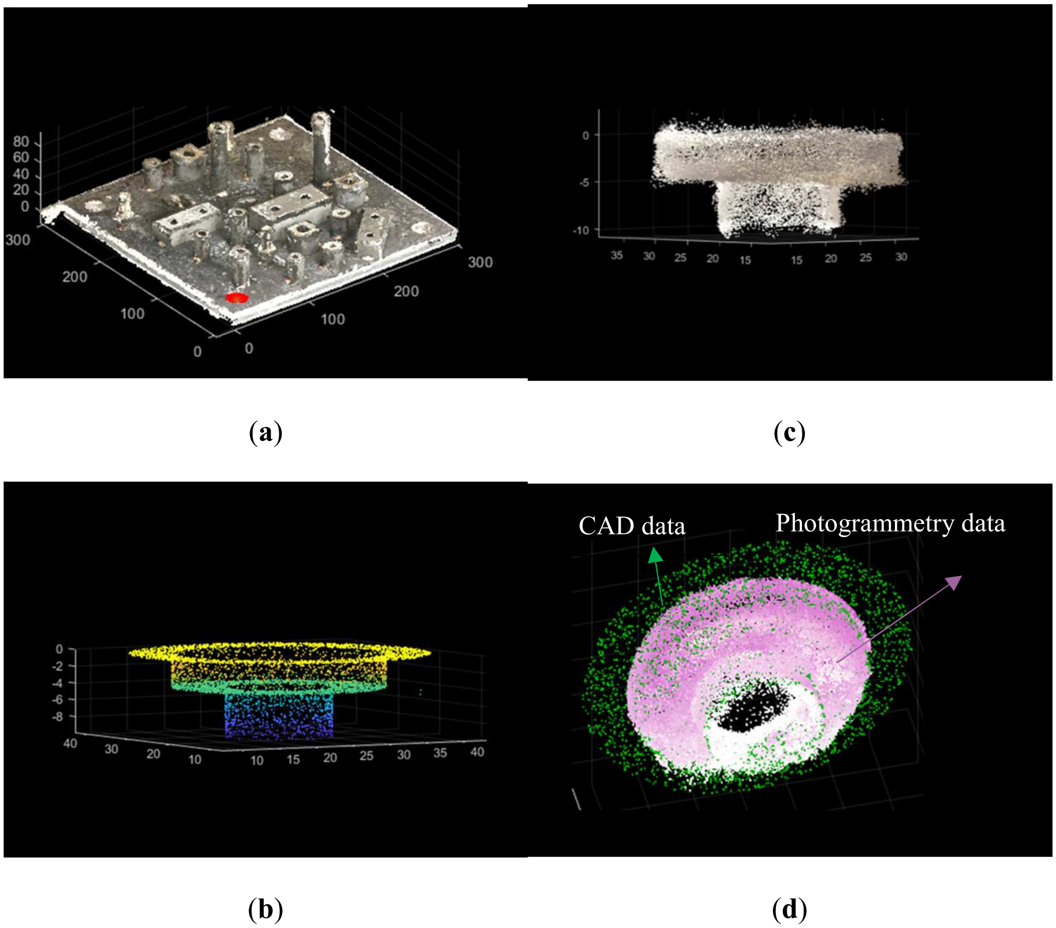


Figure 17. The fitted plane is shown on the point cloud of the whole artifact.

Next, the centers of two of the four holes in the plate are used to establish the origin and the *x*-axis. Figure 18 shows one of the corners extracted from the measured point cloud using the geometrical location on the CAD model. These extracted point clouds are then registered using a modified ICP algorithm, as shown in Figure 18 (d). For this ICP step, the *z*-axis is locked because it is defined by the already-aligned plane and only the two-translation x and y degrees of freedom are varied to align the CAD model with the data. The resulting transformation matrix then defines a best estimate of the location of the origin in the point cloud data. A similar method is used with the second hole to realize a best estimate of an *x*-axis in the point cloud data, with the DOF for ICP kept to just rotation around the *z*-axis.



(**a**) (**c**)

(**b**) (**d**)

Figure 18. Corner circles on the plate are being used for global coordinate system definition. (**a**) The circle feature shown on the data's dense point cloud, (**b**) the CAD model's point cloud of the feature, (**c**) the point cloud extracted from the photogrammetry data set of the same circle, **(d)** the two-point clouds after the ICP fine registration.

3-2-3-2 POSITION ERROR ESTIMATION

Once the coordinate systems are aligned, the CAD is superimposed with the point cloud data. The feature of interest whose position needs to be evaluated is isolated. Since we know the nominal position of the component of interest from the CAD model, the same position $\pm t$ mm is used to define a volume in the point cloud data, where the feature is likely to be. The parameter t is a chosen tolerance that reflects a maximum possible position error. This volume is cropped from the point cloud data. A CAD model of only the feature of interest is then used to locate the component's position in the point cloud data using iterative closest point (ICP) registration [25]. This process allows position errors to be assessed for arbitrary-shaped features like bosses or links, as opposed to using algorithms designed for well-defined simple geometries like cylinders or blocks.

ICP is a point cloud registration algorithm that performs a rigid transformation by searching point-to-point correspondence between two point-cloud datasets and aligning these two datasets through iteration. The rigid transformation matrix transforms the source data set into the new (aligned) dataset. After every iteration, an RMS (root mean square) error is calculated between the two points (target and source). If the RMS is above a certain threshold, the above steps are continued until convergence conditions are met [26].

The CAD feature point cloud is kept as a source (moving) dataset, and the photogrammetry data set is kept as the target point cloud to perform this registration. Before registration, the photogrammetry data set is down-sampled using grid average down-sampling by selecting the grid size equal to 0.5 mm [27]. This down-sampling averages the noise and improves the signal-to-noise ratio of the point cloud data. The grid size is chosen to have enough points for the point cloud to adequately represent the feature of interest (a post in our example) while averaging to reduce noise as much as possible. The resulting transformation matrix captures the six degrees of freedom of the position and orientation misalignment of the feature of interest and therefore contains the position error information. Around 20,000 points were kept for the target data set and 8000 for the moving point cloud.

3-3 POSITION ERROR ESTIMATION RESULTS

Once the 3-D point cloud is reconstructed and aligned to a global coordinate system, the positions of the components and features of interest can be compared with the CAD as a reference. Figure 19 shows the point cloud and the CAD registered to the same global coordinate system described above. To demonstrate the process, a CAD model is used with a shifted position of the post on the assembly (near the front), so a position error is present in the measured point cloud data and can be quantified. The Figure 19 insert is a higher magnification of the misplaced-post region.

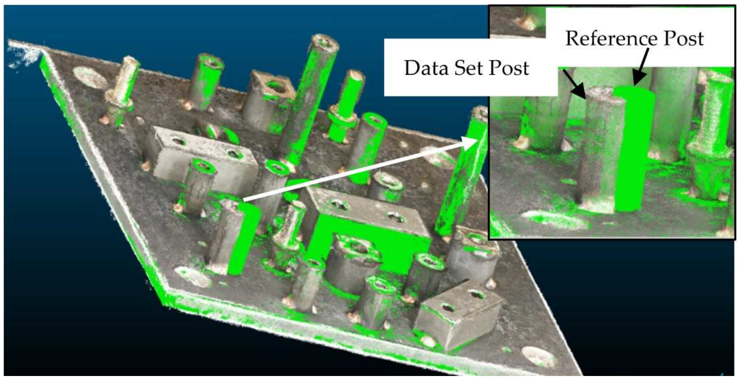


Figure 19. Point cloud superimposed with the CAD model. The insert shows a magnified misplaced component region.

The nominal position of the post in the CAD model is used to select a 3D region likely to contain the post in the point cloud data set. Figure 20 (a) shows the post isolated from the plate. Geometrical information of the CAD model is used to approximately extract the points presuming to contain the post in the photogrammetry point cloud. Then, the fine registration ICP step with the 3D point cloud of the post and the CAD model of the post is used to determine the post position error. Remember, the ICP fine registration returns 6 DOFs of the post in the 3D data and we are taking the position error measurand to be the distance between the center of mass of the two-point clouds. Figures 20 (b) and (c) show the point clouds before and after ICP registration.

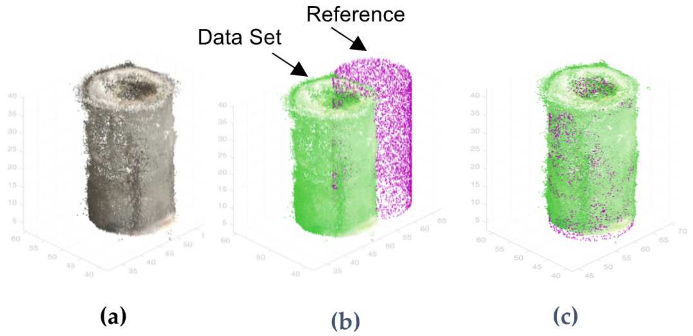


Figure 20. (a) Post isolated as a component of interest, (b) point cloud and reference data set before registration, and (c) after registration.

To check how well the registration works for large position deviations, the post in the reference was displaced from 1 to 9 mm. ICP was then performed, and the displacement between the two posts was calculated. Figure 21 shows the actual and measured position differences. The variation in this measurement falls within ± 0.2 mm of the actual, for a displacement of 1mm. The discrepancy between the two values can be explained by looking closely into the 3D reconstructed points, the limitations of ICP and manufacturing surface defects and roughness. The latter can be distinguished by taking measurements using a CMM. Noise due to the reconstruction process is affected by image acquisition processes, the reconstruction algorithms, and camera quality. In the next chapter, we explore this further.

Upon looking at the point cloud of the component closely, we can see noisy points scattered around the post; thus, it is safe to assume the discrepancies between the position estimations are due to ICP converging at local minimums for every displacement value.

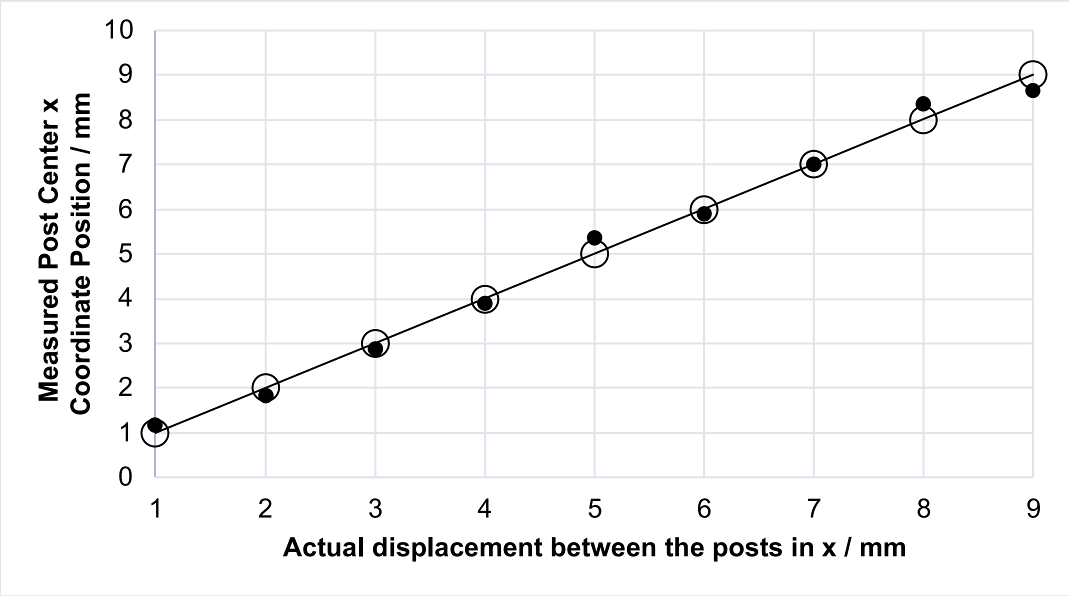


Figure 21. The actual simulated displacement error is plotted against the calculated position of the post in the photogrammetry data set.

To investigate this further and see how the point clouds behave and converge near the local minimum, another post on the plate of height 25 mm and radius 10 mm was picked and the same registration method was performed. Figure 22 shows the graph of the resulting estimate of the position deviation relative to the reference.

At 5mm post displacement from the reference, the step-by-step iteration steps of the two-point clouds while performing ICP were plotted. Figure 23 shows the result, after 8 iterations the displacement between the points in the two-point clouds is the lowest, providing maximum overlap.

To test if there indeed is a local minimum at this point, a simulation was performed where the photogrammetry point cloud was fixed at 5mm point and the reference was moved in increments of 0.1 from 4mm to 6mm. Figure 24 shows the graph of the displacement between the points on the moving point cloud and the reference point cloud at each of these distances. As seen, we see a minimum displacement at 5.3mm, which corresponds to our position estimation at 5mm in Figure 24.

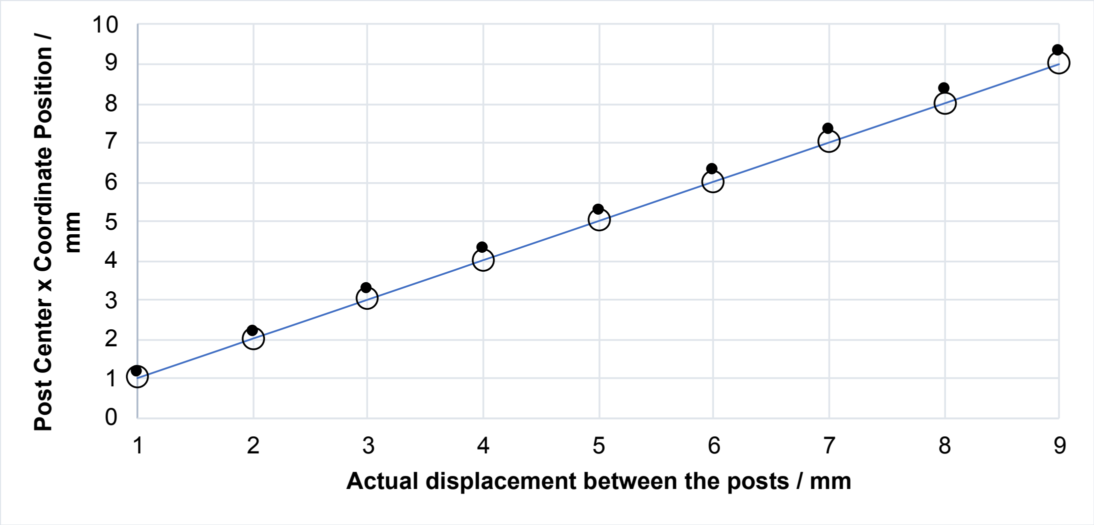


Figure 22. The actual simulated displacement error is plotted against the calculated position of the post in the photogrammetry data set.

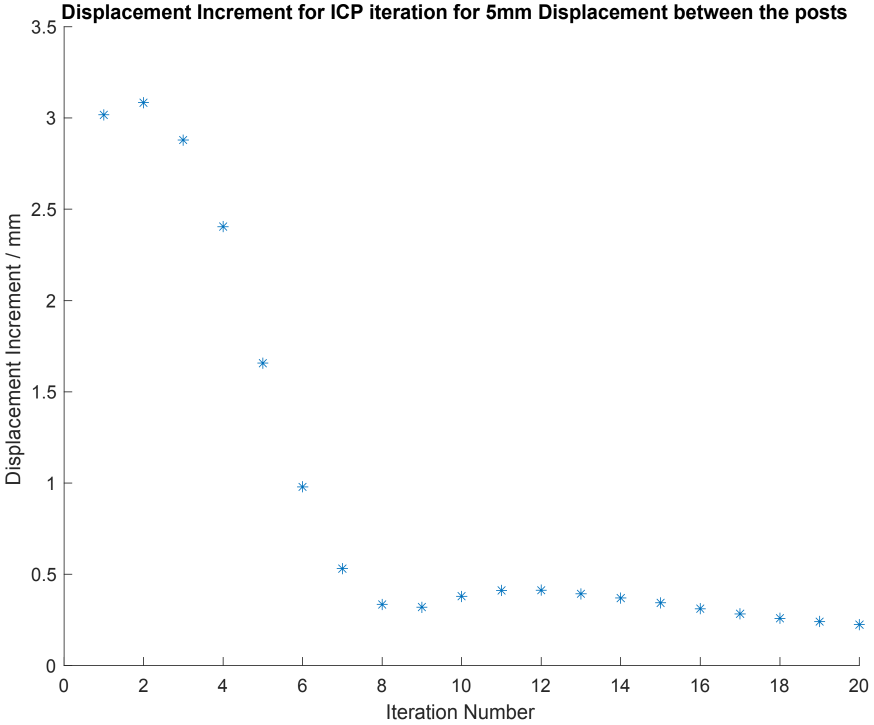


Figure 23. The displacement increment for ICP iteration for 5mm actual displacement of the post is plotted to see how the displacement between the two-point clouds is minimized while performing this registration.

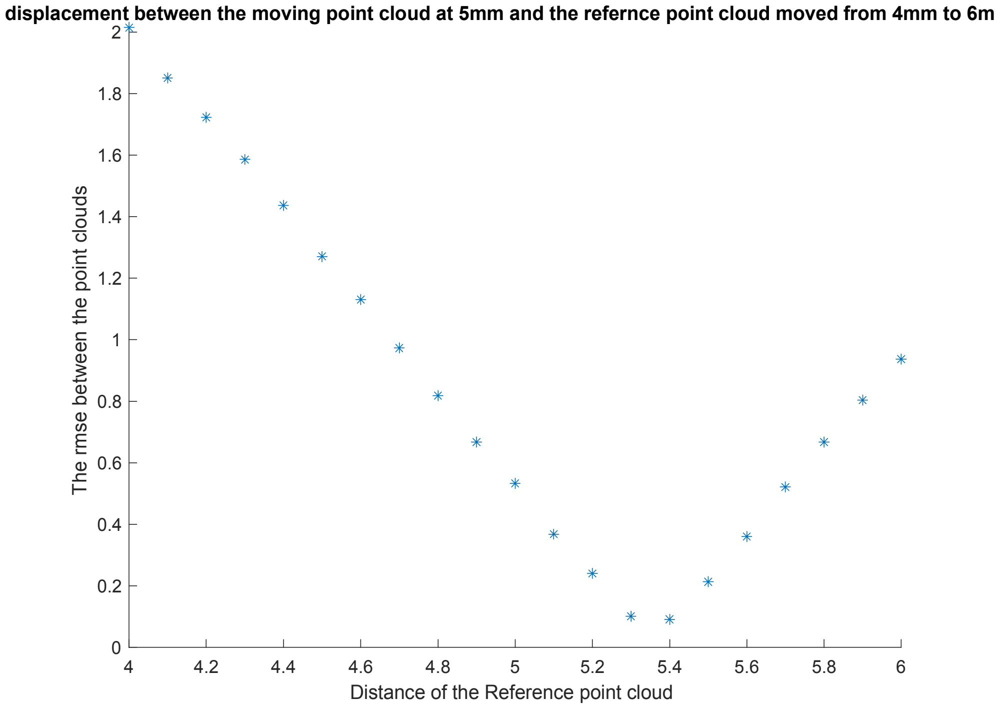


Figure 24. Reference point cloud is moved a fixed distance from the photogrammetry point cloud of the post and the displacement between the points of the two-point clouds is calculated.

CHAPTER 4 UNCERTAINTY AND CONFIDENCE IN THE MEASUREMENT

4-1 INTRODUCTION

Our measurand is defined as the displacement between the center of masses of the two-point clouds, the point cloud data from the photogrammetry and point cloud data from the CAD model. This is estimated using point cloud registration algorithms. Our goal is position error measurements of 1mm and higher with submillimeter uncertainty.

Uncertainty contributions can be divided up into three groups: i) noise and possible bias in the acquired point cloud data, ii) the actual shape of the component compared to the ideal (CAD model) component shape, and iii) data processing choices.

For the first group, the measured 3D point cloud is acquired using the structure from motion pipeline (SfM) in the photogrammetry stage. It relies on 2D images of the scene and non-optimal image acquisition conditions will affect the 3-D photogrammetry process and uncertainty in the point cloud. Good image acquisition practices like good lightning conditions, diverse camera positions, and image densities reduce point cloud uncertainty [80].

Lighting conditions and camera location choices can lead to spatial biases in the 3D point cloud. Thus, the 3D reconstruction process would be sensitive to optical properties of surfaces and lighting conditions and would be situation dependent. The contribution of these factors should be explored when the measurement process is configured for a particular application.

Random noise in the point cloud that is point-wise independent (or independent over a spatial length scale that is small compared to the dense point cloud reconstruction length scale will be present.

Some commercial photogrammetry software, such as ContextCapture [81] and Metashape [82], use global smoothing algorithms. Work done by Nealen and Igarashi [83] explains a mesh optimizing and smoothing algorithm. However, these algorithms also smooth out the finer features of the point cloud. Another technique to remove noise in the photogrammetry data is to use local feature algorithms, which are used to analyze the 3D mesh shape and the surface normal, to analyze areas of high surface roughness versus noisy regions [84, 85]. This chapter explores the impact of point-wise random noise in the point cloud on the measurand using a Monte Carlo analysis. Simulated data is used for the study and noise is added in all 3 dimensions to each point over a range of noise levels.

The second uncertainty group is the impact of the actual shape of the component compared to the ideal (CAD model) shape. This contribution will be application dependent. It will depend on the manufacturing tolerances of the components themselves to be assembled and the assembly process (e.g., welding). For the artifact used in this study, with steel components welded onto a steel plate, the investigation discussed in Chapter 3 shows position error uncertainty on the scale of several tenths of a millimeter that is consistent with this explanation. It is much larger than the point cloud random noise discussed below. The impact of point-wise random noise on the measurand likely would be the limiting uncertainty in some applications.

The third group, data processing choices, is also discussed in this chapter. This includes things like masking choices made to extract the region of the point cloud containing the component and data processing steps used to define the global coordinate system. Parameter choices for the ICP algorithm used for the registration (discussed in Chapter 3) like point cloud densities and the convergence threshold impact the measurand and should be chosen carefully to ensure these are not a limiting uncertainty contribution. ICP convergence to a local minimum can occur when the noise in the 3-D reconstructed data is high or when there are not enough geometric constraints in the point clouds (such as when the point clouds are symmetric) [86]. The investigations discussed in this chapter should be repeated when configuring the measurement to a new application to ensure ICP is configured with appropriate point cloud densities and a convergence threshold.

4-2 STUDYING THE EFFECT OF POINT-WISE RANDOM NOISE ON THE MEASURAND

Figure 25 is repeated from Chapter 2, to show the distance map between the CAD model and the photogrammetry data set. This distance map is imported from CloudCompare. We use the point-cloud to point-cloud distance computation method [124], where the Euclidean distance is estimated using the 'nearest neighbor' approach [123].

This distance map includes noise from the 3D SfM reconstruction of the point cloud, surface imperfections and texture, and geometry variations from the CAD model. We use this distance map to estimate an upper bound on the standard deviation of noise in the 3D reconstructed point cloud. This is a generous overestimation, as we are not discounting the distance map contributions from surface imperfections and manufacturing defects.

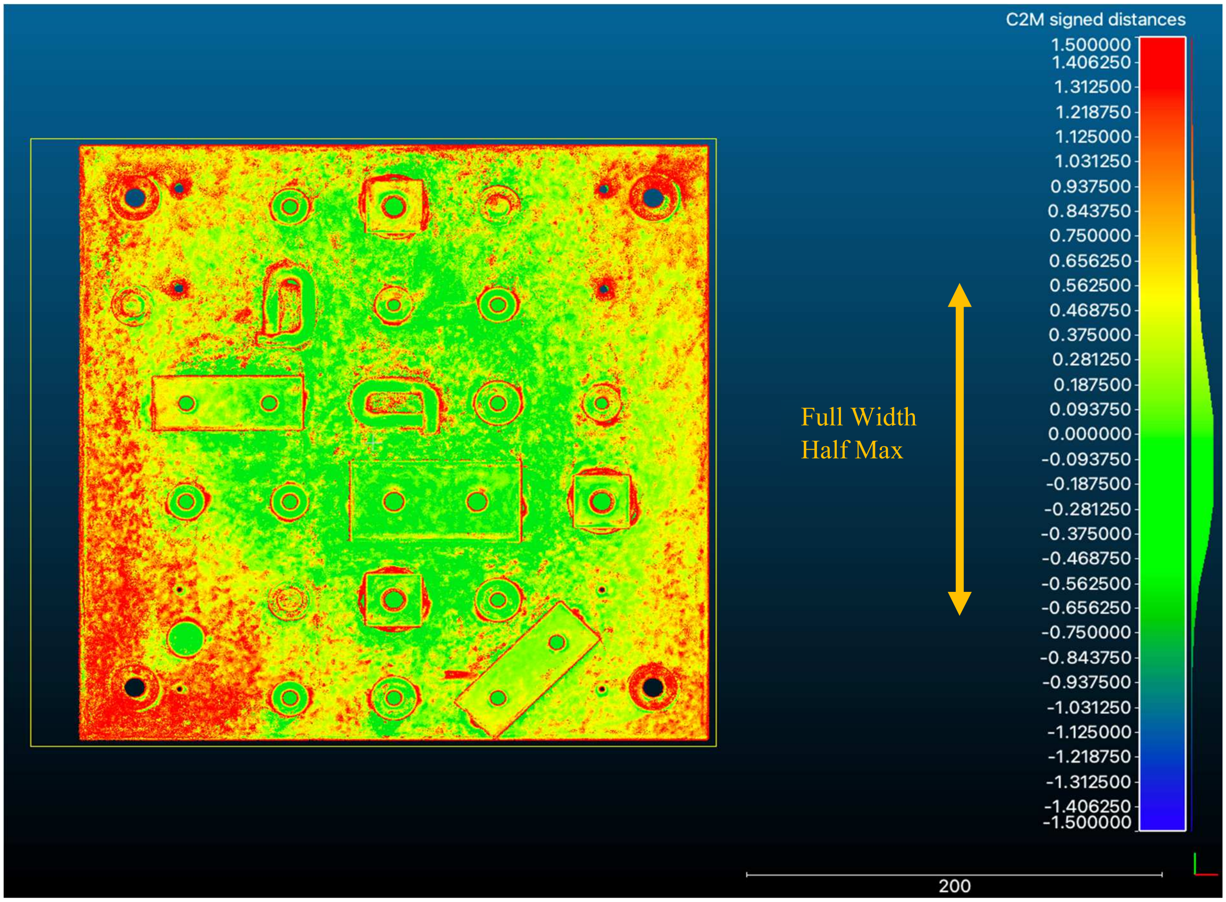


Figure 25. The distance map of the photogrammetry data relative to the reference CAD.

4-2-1 POLYWORKS AND FEATURE DETECTION

To study the effect of noise on the measurand, we begin with an analysis using the commercial PolyWorks software [87] for the position error analysis. PolyWorks inspector is metrology software that is used to study and analyze 3D data in the industry to "diagnose and prevent manufacturing assembly issues, guide assembly building through real-time measurements, and oversee the quality of assembled products by using non-contact point cloud digitizers and single-point contact-based probing devices" [87]. It is well known in the 3D data community and could be integrated into our measurement pipeline for the position error estimation step. However, as we discuss below, there are several disadvantages to PolyWorks: (i) there are noise-dependent biases in the measurand, (ii) it cannot be easily applied to complex shapes, and finally (iii) the code is proprietary and cannot be investigated or modified.

These are the steps followed for the PolyWorks study:

1. The CAD model of the artifact was imported into CloudCompare [31] as a mesh in 'ply' and 'ASCII' format. The mesh was then sampled, and around 1 million 3D points were extracted from the surface of the mesh cloud.

2. These points were then used as a base onto which random gaussian noise of different standard deviations is added to each *x*, *y*, and *z* coordinate of each point, as shown in Figure 26.

3. The post was then chosen as a feature of interest on the artifact plate and displaced 1 mm in the positive x-direction from the reference.

4. The noise was added to the *x*, *y*, and *z* coordinates of every single point on the post as a random number drawn from a normal distribution with a mean of 0 using MATLAB. Figure 26 shows the point cloud with noise added just to the post.

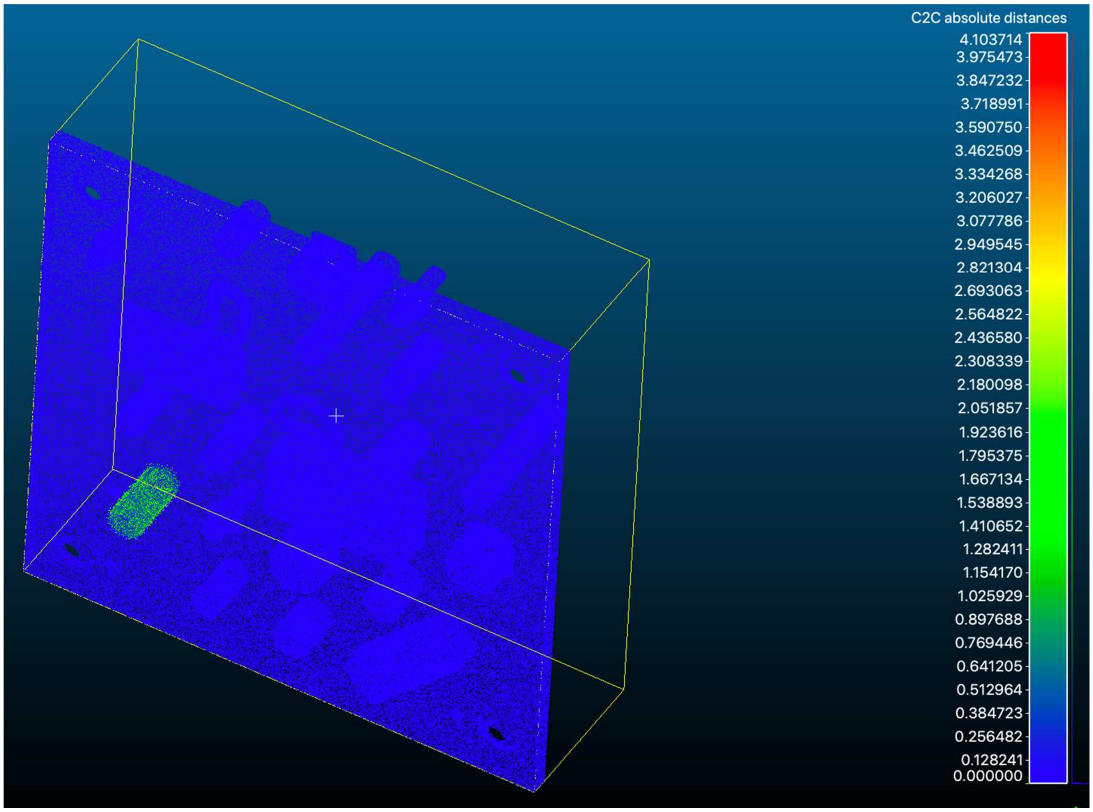
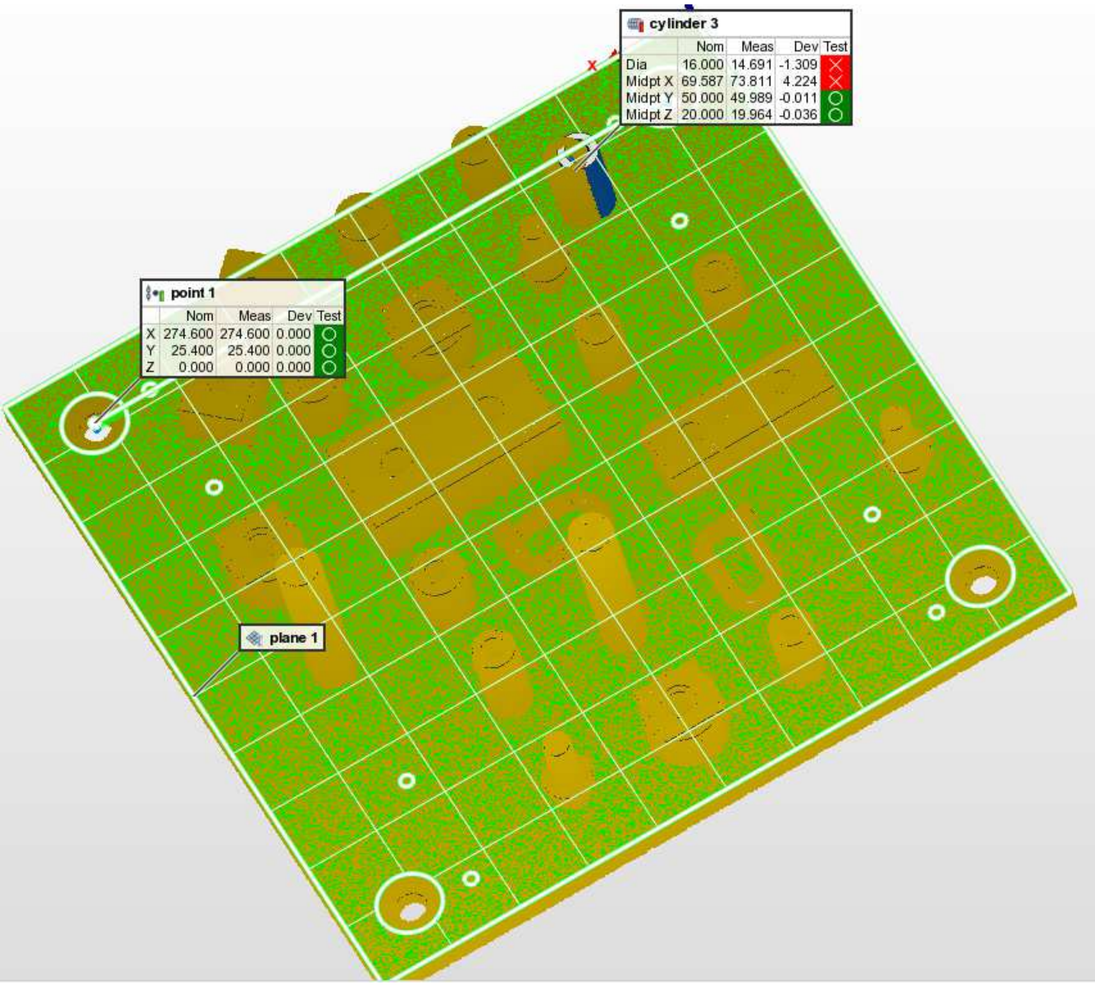


Figure 26. Post as the feature of interest with random gaussian noise added to each point.

5. For each noise level, 100 independent posts were created.

6. The global coordinate is then established in the simulated point cloud data by performing plane, axis, and center point alignment as discussed in Chapter 3, where the center point acts as the origin, the axis as the x-axis, and the plane alignment establishes the x-y plane.

7. Finally, the feature of interest was detected using feature detection for each of the 100 simulated posts. In our case, the post is detected using cylinder detection in PolyWorks and the characteristics and position of this cylinder are then compared to the reference. Figure 28 shows the cylinder detected on one of the post point clouds. Figure 29 shows the nominal (reference) point cloud and the simulated post data. The displacement between the center of the two cylinders serves as the measurand here.



cylinder 3

	Nom	Meas	Dev	Test
Dia	16.000	14.691	-1.309	☒
Midpt X	69.587	73.811	4.224	☒
Midpt Y	50.000	49.989	-0.011	○
Midpt Z	20.000	19.964	-0.036	○

point 1

	Nom	Meas	Dev	Test
X	274.600	274.600	0.000	○
Y	25.400	25.400	0.000	○
Z	0.000	0.000	0.000	○

Figure 27. Alignment of the global coordinate systems.

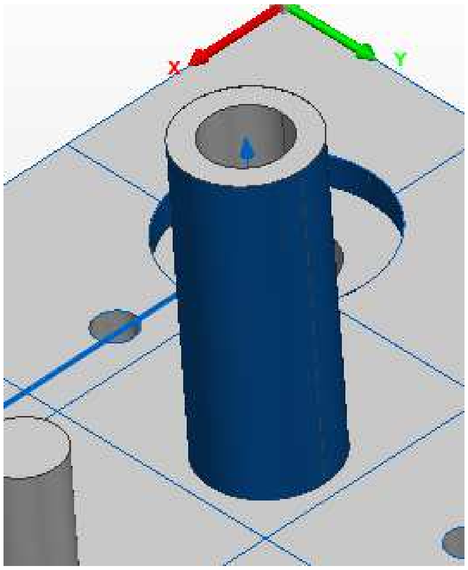
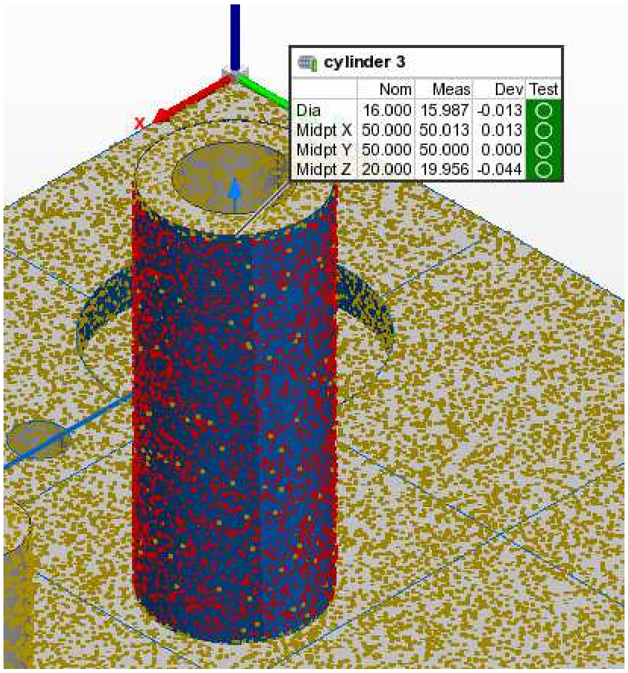


Figure 28. Cylinder detection on the measured point cloud.



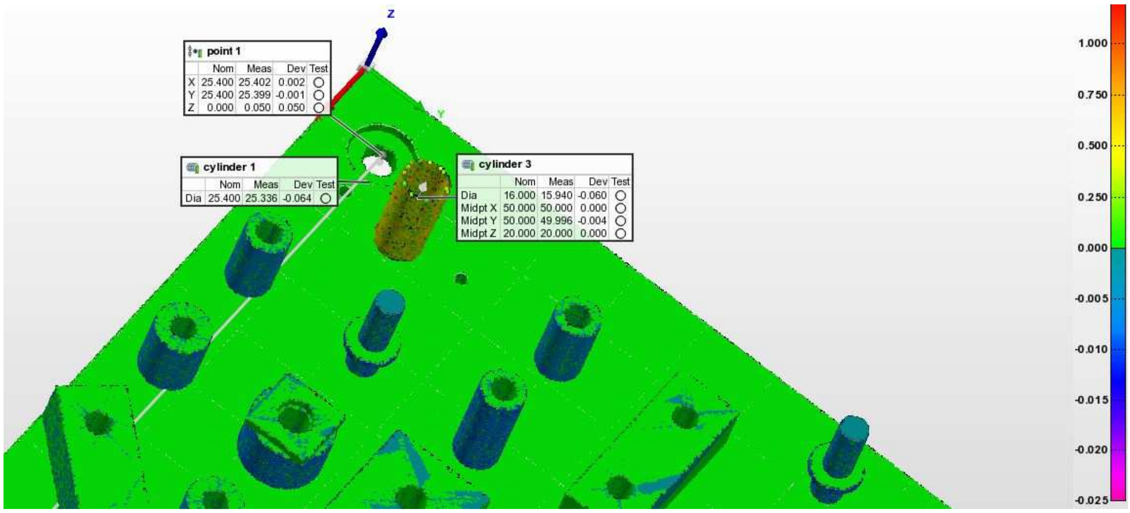
cylinder 3

	Nom	Meas	Dev	Test
Dia	16.000	15.987	-0.013	○
Midpt X	50.000	50.013	0.013	○
Midpt Y	50.000	50.000	0.000	○
Midpt Z	20.000	19.956	-0.044	○

Figure 29. Nominal and measured data for cylinder detection.

Figure 30 shows a distance map relative to the whole point cloud for the case of zero position deviation of the post. The noise on the cylinder is visible with colored points. The plate and other features show a distance of zero relative to the data set, unlike the cylinder which has the added noise.

These steps are repeated for each of the 100 simulated measurements of the post. The mean and standard deviation is calculated from the set of 100 measurand estimates and the results are plotted in Figure 31. The data points in the graph are the mean at each noise level, and the error bars are the standard deviation. The actual value of the measurand is 1mm, and the mean of the estimated data falls below this value.



point 1

	Nom	Meas	Dev	Test
X	25.400	25.402	0.002	○
Y	25.400	25.399	-0.001	○
Z	0.000	0.050	0.050	○

cylinder 1

	Nom	Meas	Dev	Test
Dia	25.400	25.336	-0.064	○

cylinder 3

	Nom	Meas	Dev	Test
Dia	16.000	15.940	-0.060	○
Midpt X	50.000	50.000	0.000	○
Midpt Y	50.000	49.996	-0.004	○
Midpt Z	20.000	20.000	0.000	○

Figure 30. The distance map of the reference relative to the measured point cloud.

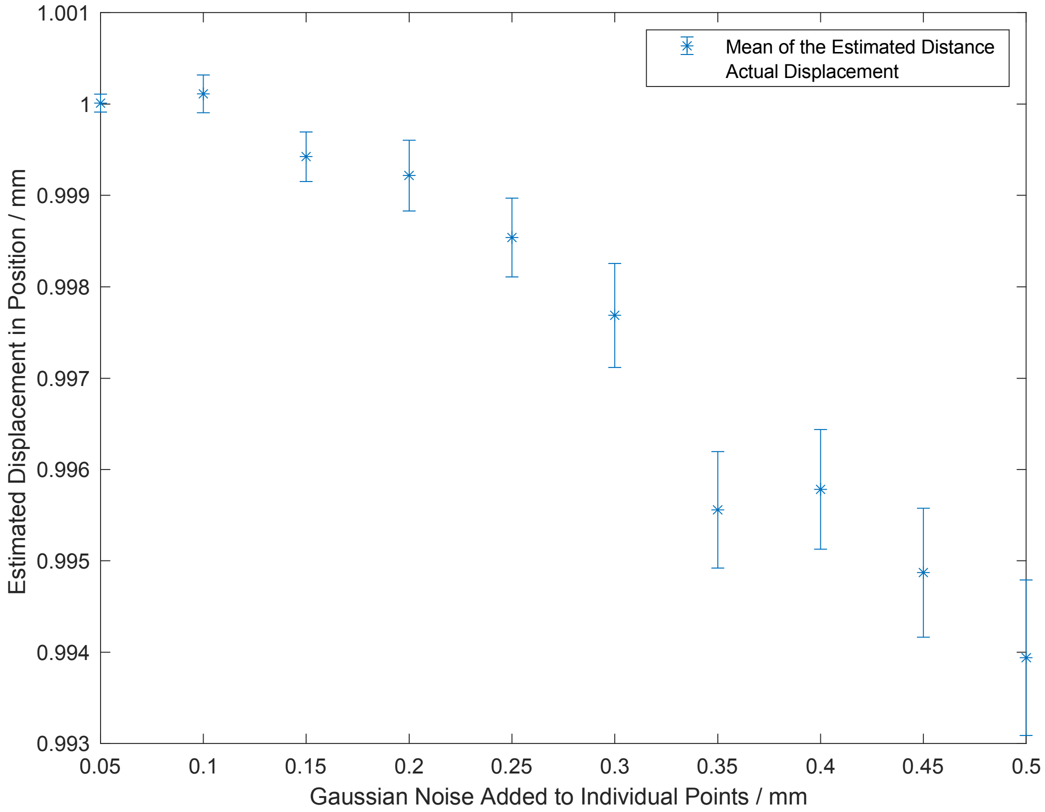


Figure 31. Results of the Monte Carlo study using PolyWorks cylinder detection to estimate the measurand for simulated post data with different noise levels. The horizontal axis is the standard deviation of the Gaussian noise added to each *x*, *y*, and *z* coordinate in the simulated data.

The measurand here is estimated from PolyWorks' cylinder fit to the point cloud of the post which we expect works differently from our approach using ICP discussed in the previous chapter. The standard deviation of the measurand increases with the noise level, as expected. However, we see a bias in the measurand on the micrometer-scale that increases with the noise level. The details of the PolyWorks feature detection algorithm are unknown aside from the general description that the feature of interest is selected from the CAD model and then the software searches for this feature in the nearest cluster of points, trying to fit this model (cylinder) onto the point cloud data. Because the algorithm is proprietary, all we can do is speculate that perhaps the convergence threshold is on the micrometer scale and the optimization algorithm stops prematurely or gets stuck in a local minimum as it moves the reference closer the data point cloud, always stopping at a position closer than the actual best-fit position. Even though the bias is small, not knowing the cause of the bias precludes an understanding of how the bias will track with component geometry, component size scale, point cloud densities, etc. This unknown, in addition to the difficulty of using PolyWorks for general complex components, motivated the use of ICP for the position error estimation instead.

4-2-2 ICP REGISTRATION UNCERTAINTY STUDIES

We performed a similar Monte Carlo simulation in MATLAB using ICP for the position error estimate instead. The same steps are followed as outlined in the previous section with random Gaussian noise added to the *x*, *y*, and *z* coordinates for each point in the point cloud over a range of noise levels. The noise level range was significantly increased for this study to capture the large-scale trends. The distance between the reference post and the simulated posts with noise was fixed at 1 mm. The ICP algorithm was used to register the two-point clouds together for each of the 100 data sets at each noise level considered. The results are shown in Figure 32. To graph these results (and compare to a study discussed in the following section) the graph shows the mean and standard deviation (error bars) of the 100 measurand estimates at each noise level as a function of the noise level of the center of mass (the geometric center) of the post. For a given noise level on the x, y, and z coordinates with standard deviation σ_p for every point making up the cylinder (e.g. 10,000 points), the noise level on the geometric center of the post is significantly lower. The geometric center of the post is an average of the n points in the post point cloud and therefore the uncertainty on the center is approximately $\sigma_c = \sigma_p/\sqrt{n}$. The greater the number of points, the smaller the uncertainty in the coordinates of the center. This must be kept in mind when comparing the PolyWorks and Matlab studies.

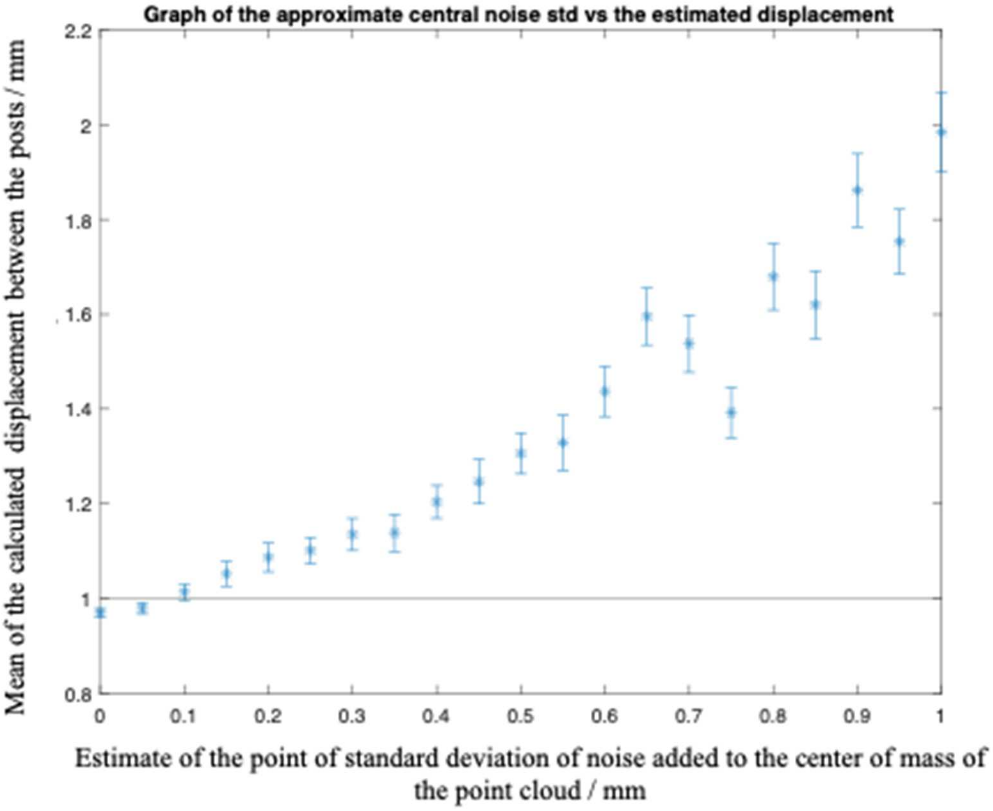


Figure 32. Investigating the effect of noise on the center of mass of the point cloud on the measurand calculated using 100 ICP runs for each noise level. The x-axis represents an estimated standard deviation

of noise added to the center of mass of the point cloud and the y-axis shows the mean displacement between the two posts.

To compare these results to the worse-case noise level for the experiment with $\sigma_p \sim 0.4\ mm$ (estimated from Figure 25), for the $10{,}000$ points making up the post, that would mean $\sigma_c \sim 0.04\ mm$. At that noise level in the graph in Figure 32, the simulation shows a bias of ~ $0.03\ mm$. The small $0.03\ mm$ bias at the low noise levels that is less than 1 mm can be reduced using a lower ICP convergence threshold than the default parameter and by down-sampling the noisy or measured data set compared to the reference. We explore these issues in Section 4-4. The bias larger than 1 mm that is progressively larger with increasing noise levels is due to the Raleigh statistics in the large noise limit of a measurand like a displacement that cannot be negative. This is discussed in the following section.

4-3 STATISTICAL DISTRIBUTION OF NOISE

To better understand the presence of the bias in the large noise limit, we study the research presented by Morse and Voelcker [121]. The article goes over the assumption that gaussian statistics can be used for some dimensional data and argues for using Rayleigh statistics for data where the measurement value approaches the noise level. It proves the method's validity by exploring the position tolerance of a punched hole. The probability of having the estimated position measurand value r is given by

$$P(r) = \frac{re^{-\frac{(r^2+A^2)}{2\sigma^2}}}{\sigma^2} I_o(rA/\sigma^2)$$

$$for\ r \geq 0\ ,$$

where A is the actual position error of the punched hole. The variable I_o is the zeroth order Bessel function of the first kind. The variable σ is the gaussian standard deviation.

We follow this Rayleigh statistical distribution of the measurand around the actual value (A) with noise in the experimental data presented by σ_R, which is the Rayleigh standard deviation. The variance can be related to σ, the normal standard deviation, by the formula below.

$$\sigma_R^2 = \sigma^2 \left(2 - \frac{\pi}{2}\right).$$

The distribution approaches a normal (gaussian) distribution when the noise becomes significantly smaller than the offset value.

This can be illustrated with a simulation. The distance between two points in 2D space was calculated. Some 10,000 points were simulated, and random noise of a standard deviation of $0.3mm$ was added to one of the points. Three such experiments were done where the true distance (offset) between the points was changed. Figure 33 shows the basic principle.

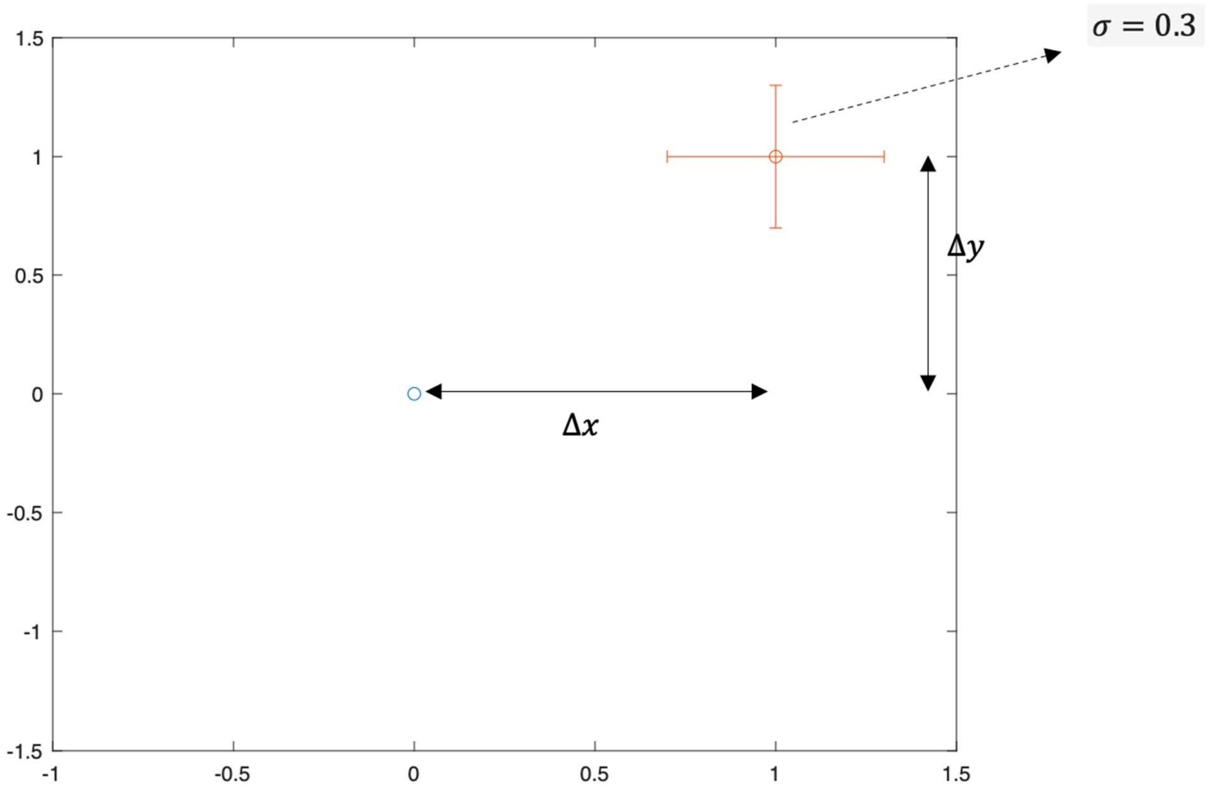


Figure 33. The basic principle of the statistical distribution simulations.

Figure 34 shows the distribution where there was no offset between the two points, and the distance was calculated. The results show a clear bias where A is the actual value of the measurand, and A' (the mean of the distribution) is the apparent value.

Figure 35 shows the histograms for an offset of 1 mm. Increasing the offset such that the actual position deviation is much larger than the noise in the dataset gives a more normal (gaussian) distribution, with A and A' converging.

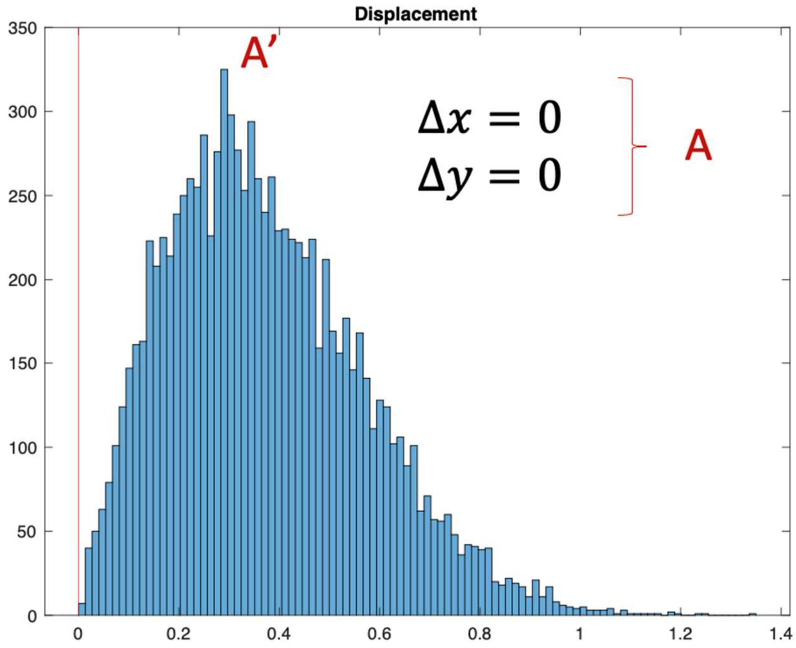


Figure 34. Histogram of 0 offsets between the points, the x axis shows the distance between the reference and simulated noisy points in millimeter and the y axes shows the number of points in each distribution.

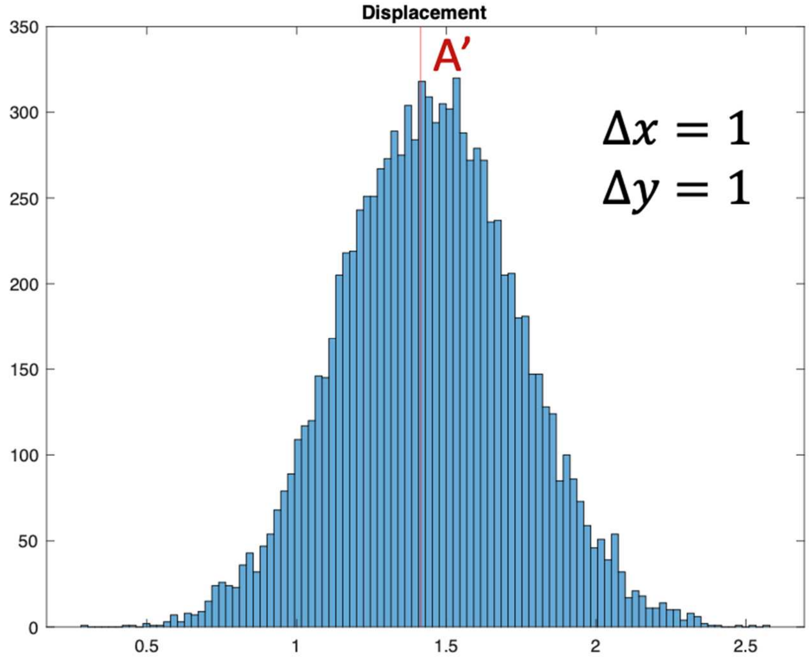


Figure 35. Histogram of 1 mm offset between the two points. The red vertical line shows the actual displacement, as seen the mean of the distribution falls on the red line.

For our application, the random noise due to the photogrammetry 3D reconstruction step stays well below $0.3mm$, and it is safe to assume the effect of this bias is not significant as we reach a more Gaussian distribution. Removing this bias for applications where the noise levels are comparable to the displacement becomes a necessary step.

Our target goal is to measure the position displacement values of $1mm$ and higher; for this, we can go up to $0.3mm$ of noise on each point in the point cloud for the Rayleigh bias to not significantly affect the uncertainty in our measurand.

4-4 IMPROVING ICP-BASED REGISTRATION RESULTS

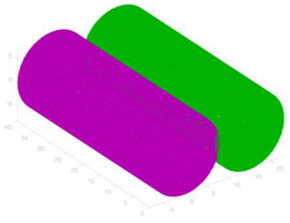
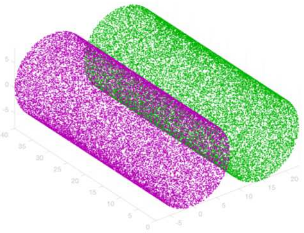
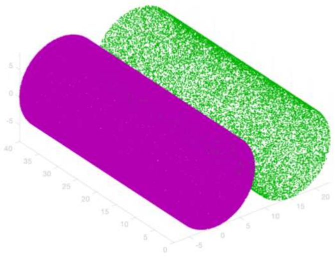
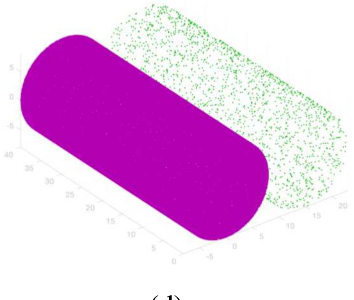
4-4-1 DOWN-SAMPLING

Apart from the noise present in the 3D reconstructed point clouds, uncertainty in the measurement can also be attributed to the ICP registration method we are using for our application. To explore this further, we performed simulations using two-point clouds with no noise.

The point cloud of a cylindrical post was used to investigate this. We tried different combinations of point cloud densities to see how this affects the final measurand value. Table 2 shows the different combinations.

Figures 36 (a) – (d) show the results of these simulations. Figure 36 (a) has both dense point clouds for reference and the moving point cloud. This gives a reasonable estimation of the measurand for the case of no noise. However, it takes much longer to perform the registration. Using a sparser point cloud for the moving data set solves this problem, as shown in Figures 36 (c) and (d). The graph in (d) has more uncertainty present in the result than the case for (c), implying an optimal ratio between the dense and sparse point clouds represented by the conditions used in (c).

Table 2: Different densities of point cloud registration simulations.

Simulation	Number of points in the reference point cloud	Number of points in the moving point cloud
 (a)	240,965	240,965
 (b)	24,099	24,099
 (c)	240,965	24,099
 (d)	240,965	2,411

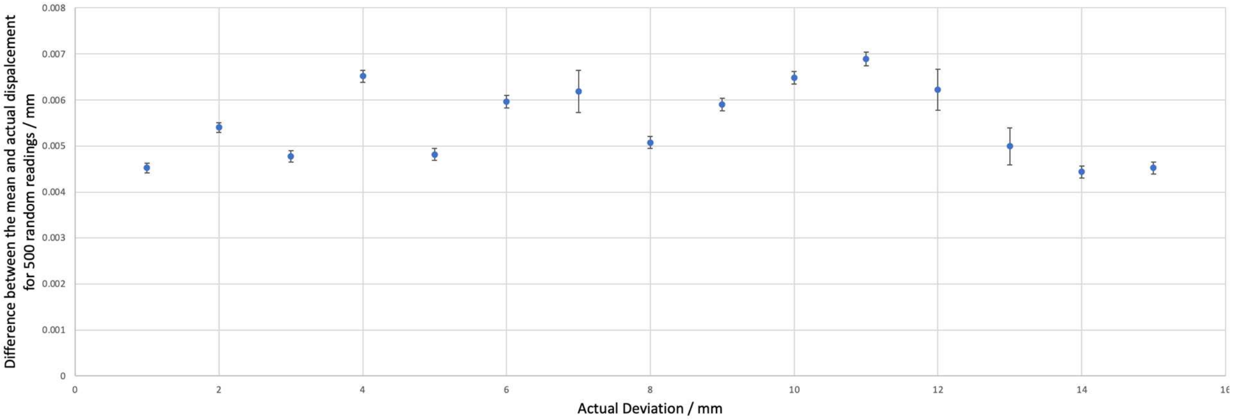


Figure 36 (a) Results of ICP registration for 500 data sets, the noise is kept to zero and the density ratio of reference and measured simulated data set is kept as Table 2 (a), with two dense point clouds. In the graph, the actual displacement vs. the estimated displacement is plotted.

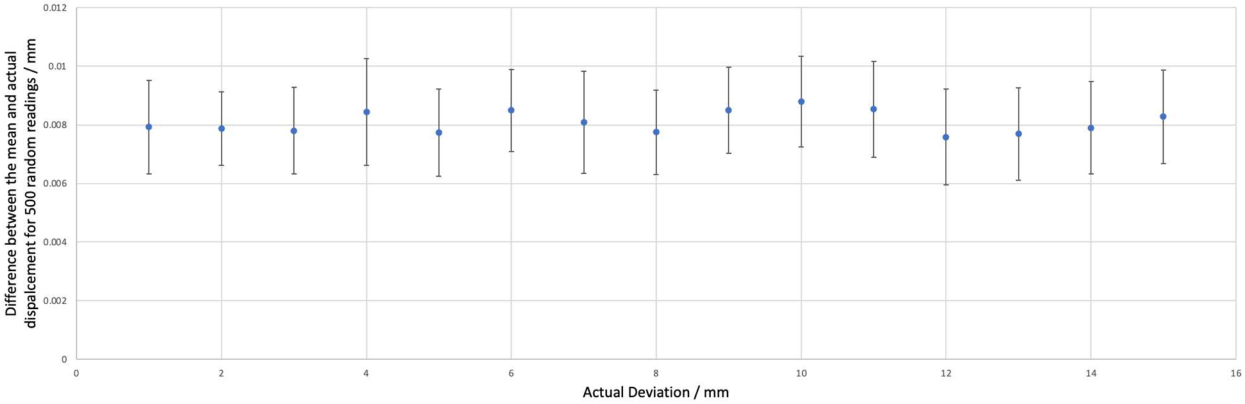


Figure 36 (b) Results of ICP registration for 500 data sets, the noise is kept to zero and the density ratio of reference and measured simulated data set is kept as Table 2 (b), with two sparse point clouds. The graph of the actual displacement vs. the estimated displacement is plotted.

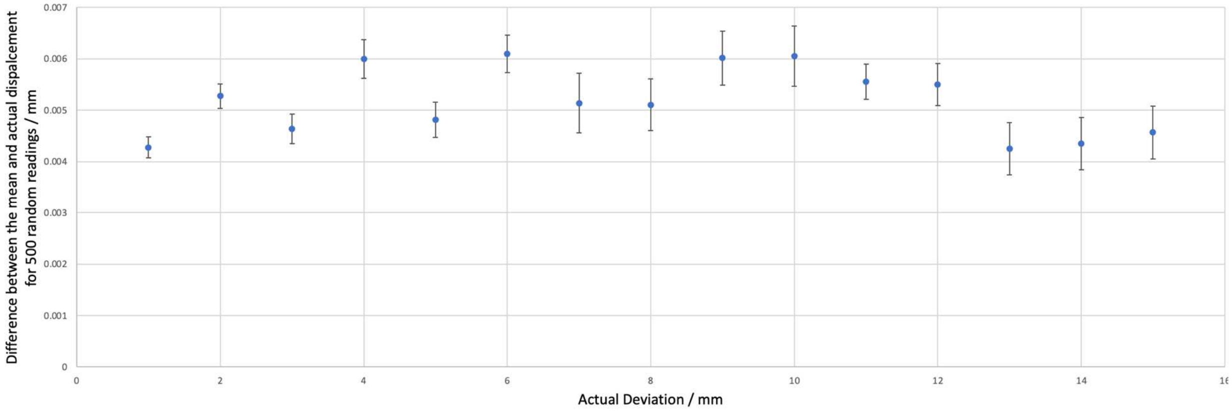


Figure 36 (c) Results of ICP registration for 500 data sets, the noise is kept to zero and the density ratio of reference and measured simulated data set is kept as Table 2 (c), with one sparse and one dense point cloud. The graph of the actual displacement vs. the estimated displacement is plotted.

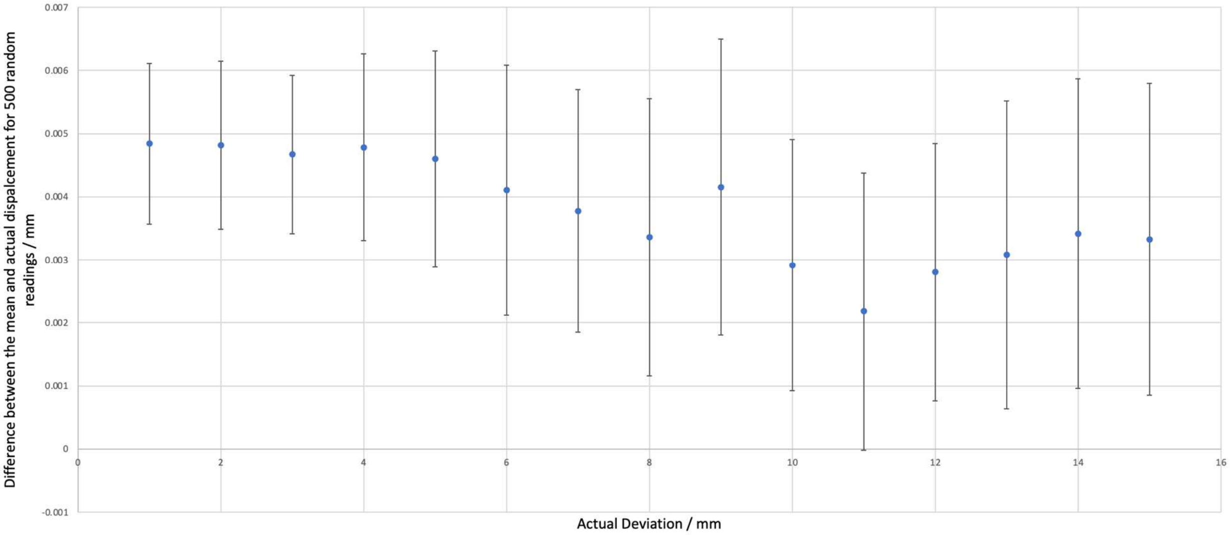


Figure 36 (d) Results of ICP registration for 500 data sets, the noise is kept to zero and the density ratio of reference and measured simulated data set is kept as Table 2 (d), with two sparse point clouds. The graph of the actual displacement vs. the estimated displacement is plotted.

4-4-2 ICP CONVERGENCE THRESHOLD

The iterative closest point (ICP) algorithm works iteratively by minimizing the distance between the two sets of point clouds. It converges when the maximum number of iterations has been reached or when the absolute difference in translation and rotation of the point clouds estimated in the last three consecutive iterations reaches a certain threshold. ICP uses two thresholds, the distance translation vectors and the angular difference in degrees.

We take the conditions explored in Section 4-4-1 and Figure 36 (c) as an example of the default threshold, which was kept at a distance threshold of 0.01mm and an angular threshold of 0.05 degrees. The iterations stop well before the maximum number of iterations (20) is reached, as shown in Figure 37. This was done for the case of 1mm displacement and point cloud densities of Figure 36(c), i.e., reference posts of 240,965 and 24,099 points.

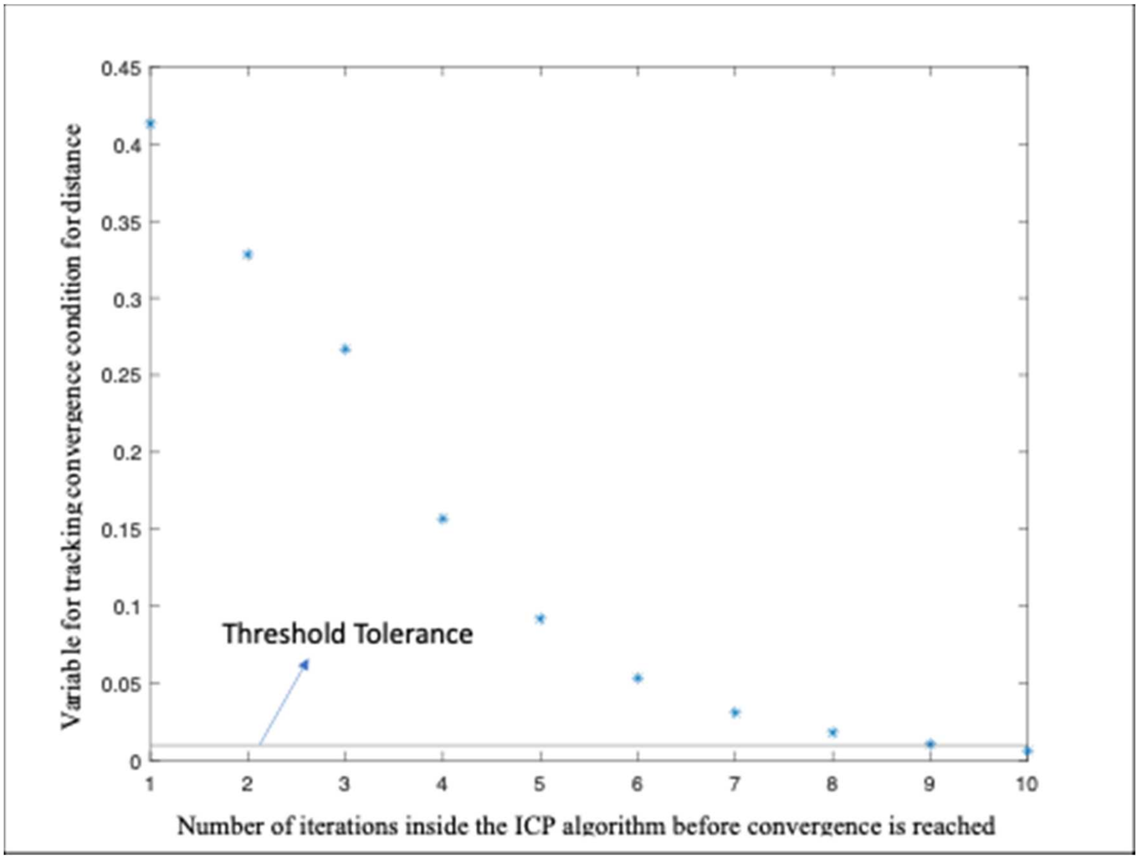


Figure 37. In the graph to investigate the convergence condition of the ICP algorithm, the threshold for distance is set to 0.001mm.

Figure 38 is a graph of distance estimation using ICP for varying distance thresholds. As expected, ICP stops the iterations and converges to a biased value on the same scale as the threshold value itself. We repeated the optimal sampling conditions in Figure 36 (c) with 0.0001 mm for the ICP distance threshold and 0.0001 degrees for the angular threshold and the results are shown in Figure 39. As expected ICP converges at positions biased from the true position error on the order of 0.1 μm, the same order as the distance threshold setting.

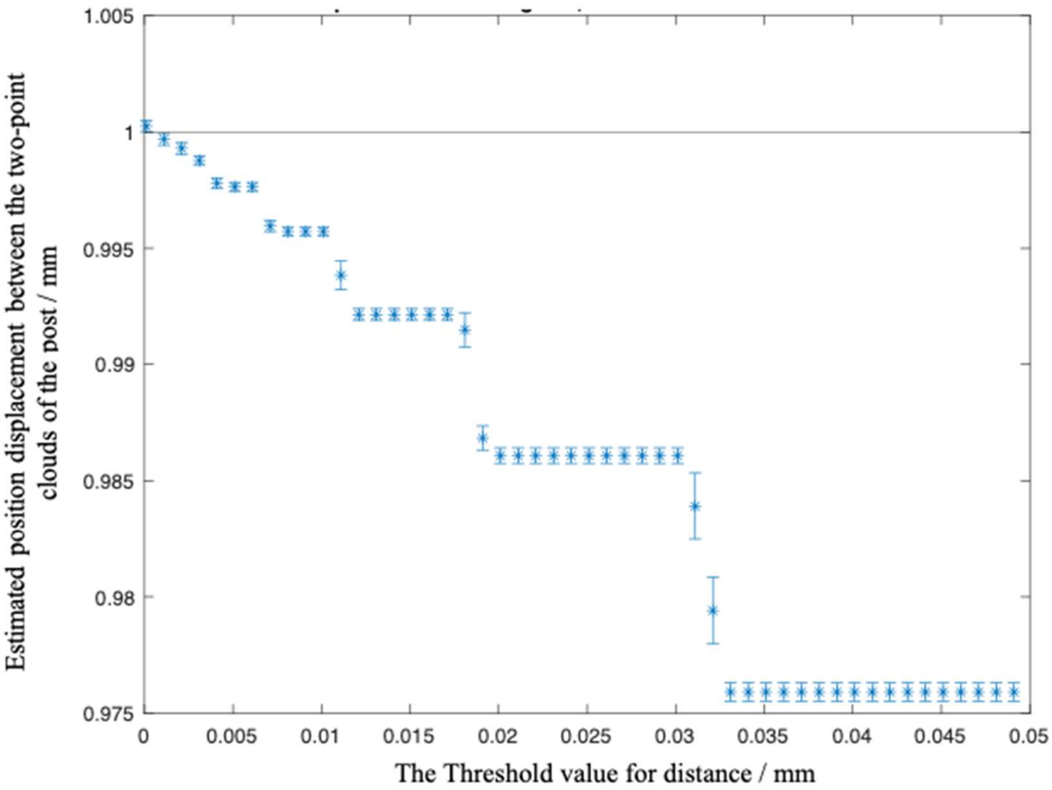


Figure 38. Effect of changing the ICP distance threshold on the measurand for a position displacement of 1 mm.

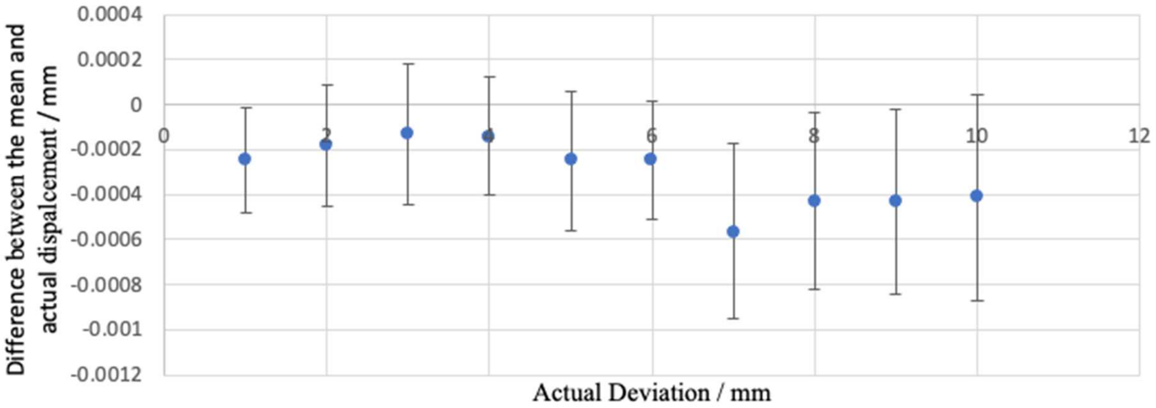


Figure 39. Estimating the difference between the actual and estimated post position using ICP, low threshold levels of 0.0001 for distance and angular tolerance.

4-5 SUMMARY

In this chapter, we explored limiting factors for our position displacement calculations of the post and how they affect our application. The target position errors (in assembly line on the component of interest, e.g., post in our case) that we want to measure are 1 mm and higher with an uncertainty of, ideally on the order of 0.1 mm.

For no random noise and optimizing the ICP parameters by using low ICP thresholds and optimal point cloud densities of the reference and the measured data set, the bias in estimated position errors can be reduced far below our uncertainty goal of 0.1 mm. For example, a displacement of 1 mm of the post (of radius $8mm$, containing around 240,000 points in the reference and 24,000 points in the measured data set), the measurand is shown to have a mean bias of 0.2 μm due to the distance threshold parameter setting of 0.1 μm

We also studied the effect of the Rayleigh bias. The target measurand values for us are 1 mm and higher. As shown in this chapter, for a noise level of 0.3 mm on the geometric center coordinate of the post, the Rayleigh bias will not impact position error measurands of 1 mm and higher. For example, a 10,000-point cloud measurement of a post with a 0.3 mm noise level on the geometric center coordinates translates to a 30 mm noise level on each point, which is well above experimental noise levels. However, for position errors lower than this, the effect of the Rayleigh bias cannot be ignored.

With regards to point-wise random noise, for a noise level of $0.4\ mm$ which is much higher than that present in our photogrammetry data, as discussed in Section 4-2 and shown in Figure 32, leads to an uncertainty in the measurand below $0.1\ mm$ even with non-optimal ICP parameter conditions.

The investigations in this chapter show that the measurement variation on the scale of several tenths of a millimeter observed with the experimental data discussed in Chapter 3 cannot be explained by point-wise random noise, data sampling conditions, or ICP parameter choices. The experimental variation discussed in Chapter 3 is most likely due to the actual component shape compared to the CAD model, surface imperfections, noise due to photogrammetry processes. Errors due to manufacturing defects are application specific and will depend on the manufacturing tolerances set on the components themselves. Noise added on the 3D points due to SfM reconstruction is complex to analyze. Different steps, such as sparse and dense reconstruction and image feature detection techniques, contribute to noise and distortions in the point cloud. More work is needed to estimate a comprehensive combined uncertainty.

CHAPTER 5 AUGMENTED REALITY AND VISUALIZATION IN METROLOGY

5-1 INTRODUCTION

Mixed reality encompasses both augmented reality and virtual reality. Augmented reality (AR) adds to the real-world scene by overlaying virtual objects on top of the user's visual environment [96], allowing the user to manipulate and interact with the real-world environment [97]. In comparison, virtual reality [VR] immerses the user in a virtual environment where the real world doesn't exist. Examples of pure VR applications would be VR games and virtual tour guides. A famous example of an AR application would be the Pokémon game, where users used their cell phones to locate and capture different virtual objects overlayed on the live videos on the cell phone.

The term augmented reality was coined in the 1990s. The term was developed by scientists at Boeing corporation when they invented an AR system to assist workers in wiring their harnesses [92].

This chapter goes over our use of AR in assembly quality inspection in industry and the conventional alternatives. We present a prototype model and application created for this purpose and demonstrate it on our artifact.

5-2 VISUALIZATION METHODS IN THE INDUSTRY

This section covers the scientific literature and augmented reality applications in the manufacturing industry. With the advent of smartphones and tablets with built-in cameras and processing units, AR is becoming increasingly popular in modern-day manufacturing industries.

5-2-1 TRADITIONAL METHODS OF VISUAL INSPECTION

Traditionally data acquired from the quality check process, where quantitative analysis is done, is in the form of tables and figures. Commercial software such as GOM Inspect [122] and PolyWorks are increasingly utilized. The evaluation of the data is done with an inspector looking over the data table and locating the error region.

Additionally, commercial software often offers distance maps (color maps indicating areas of high deviations) and values of geometric dimensioning and tolerancing (GD&T) displayed on a computer screen during inspection. The use of AR to overlay the quantitative inspection information onto a live video stream streamlines the human interface, improving efficiency by directly visually connecting the information to the physical location on the part.

5-2-2 AUGMENTED REALITY APPLICATIONS IN INDUSTRY

The term 'Industrial Augmented Reality' was coined by Georgel [94] to describe the use of AR in industrial manufacturing, design, inspection (assembly and maintenance), and training applications.

The application of AR in industry ranges from maintenance apps and training modules to product design [88]. In 1968 Ivan Sutherland and his students created the first VR and AR head-mounted system [90]. The system was primitive and heavy but paved the way for future AR and VR applications. With the invention of mobile technology from the 1970s onwards, small computers could now be worn all the time.

There are three ways to use augmented reality. The first one employs a video-see-through technology where AR is displayed on a live video feed or digitized images. Another way is to use an optical system that displays AR information through mirrors and lenses in a head-mounted display. The third way is to overlay the virtual information onto the real world using projective technology. This third option is a new field of research and works like plasma technology and is underway [93].

Among these, video see-through technology is the most convenient, flexible, and cost-effective to construct. The video is already digitized, so overlaying AR information is straightforward, and data and objects can be added and taken out when needed. Similarly, the contrast and color of the overlayed virtual information can be adjusted to match or contrast that of the video feed, whichever is desirable.

However, video feed AR technology also offers some disadvantages, such as low resolution of the video feed itself and the fixed focus condition in most video feed imaging systems like a cell phone or tablet PC. This contrasts with head-mounted display technology where the focus can be adjusted [88]. Nevertheless, for our applications and in keeping with the view to make the technology accessible and cost-effective, we are creating a video-feed AR technology in the form of an app for mobile technology.

Thanks to the advent of open source and freeware software and app toolkits, AR applications have seen a big boom. Advanced camera systems have allowed us to accurately predict the position of objects in real-time, making AR technology much easier to develop.

There are several applications for manufacturing that currently use video feed AR technology. For example, industrial maintenance applications for AR exist. Maintenance processes are performed by trained personnel and an AR interface is used to organize the steps into tasks to establish uniform and thorough procedures to aid maintenance processes. Vignali [95] explains one such application where the AR application mainly supports the maintenance inspector.

Video feed AR applications like FARO [14] are used to visually inspect the industrial environment, such as a production plant or an automotive car, to perform a quality inspection. However, this application only provides a visual qualitative assessment and leaves the decision up to the user on whether the object passes the quality check or not.

5-3 AR-BASED PROTOTYPE MODEL FOR ASSEMBLY

This section presents the last step to our overall quantitative inspection prototype model. This step imports the component's position error information retrieved from the previous steps in the pipeline and then effectively displays this quantitative information to the user. This display step is the use of AR to overlay the position error estimates of components failing inspection along with the 3D CAD model of the component at the design location in real-time on a live video feed of the part. We utilize Vuforia [28] in conjunction with Unity [29] to create this application and the details are described in the following sections. Guide views are made using the CAD model. The guide views are used to search the live video feed for the part in the real world. Once the part matching the guide view is identified in the live video feed, the CAD model of the component(s) (features of interest) for which the position error has been measured (e.g., a post) is placed in the live feed of the scene beside the real-world post in the scene. A label is added to the live feed to display the measured positioning error.

5-3-1 COMPUTATIONAL TOOLS AND ARCHITECTURE

We utilize Unity and Vuforia to make this prototype application. Unity is a famous game engine, offering an innovative platform for creating augmented reality applications. The Vuforia library is used in conjugation with unity. When combined with computer vision techniques, an augmented reality software development kit (SDK) can be used to create a complete the AR application.

5-3-1-1 MODEL TARGET GENERATOR

The first step is to create the guide view that is used to search the live video feed for the part in the real world. This is done by using a model target generator to create an object tracking class. The object tracking class for the real-world model target tracking for our application is generated using the CAD model of our system (shown in Figure 40). Figure 41 shows the guide view created from this model target using the target model generator in Vuforia. Using a phone camera or tablet, this guide view is then used to locate and track the object in the live video feed in the real world. The guide view should contain unique features to distinguish the object in the real-world environment.

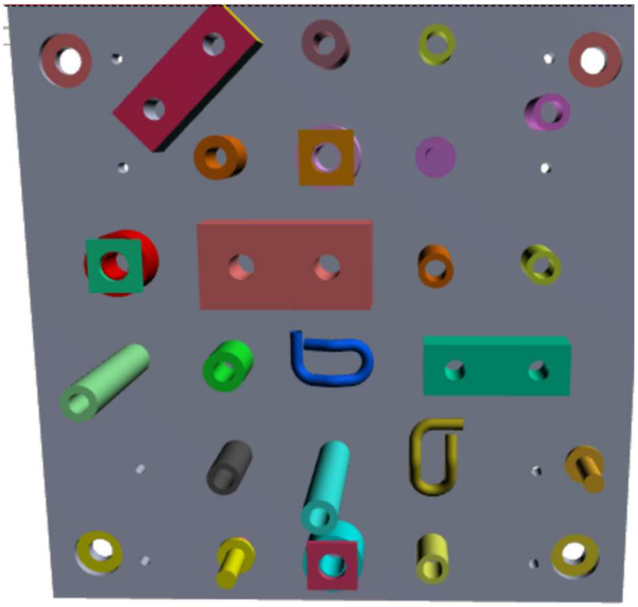


Figure 40. CAD model of the artifact used to create the object view class.

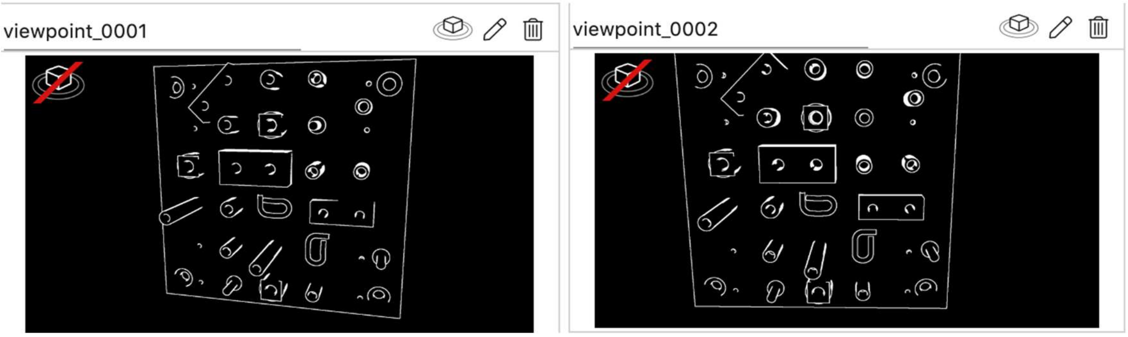


Figure 41. Guide views created from the CAD model at different angles.

We created two guide views, one of the whole artifact and the second of just one block on the plate as a component of interest (Figure 42). The goal was to test the level of complexity needed in the guide view to locate the part in the live video feed.

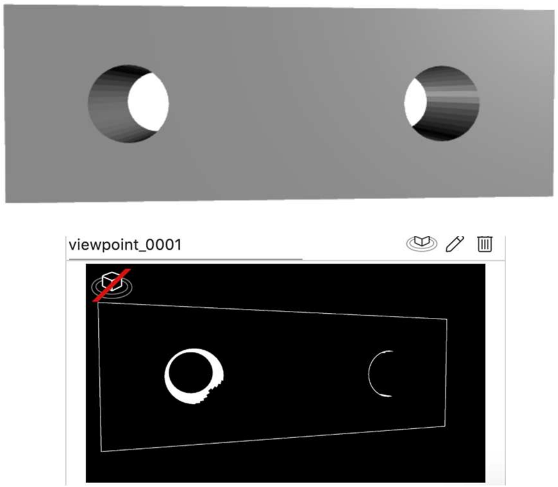


Figure 42. Guide views created from the CAD model of the block

5-3-1-2 DETECTION IN LIVE VIDEO FEED

First, we show the results with the full artifact as the guide view (Figure 41) used for the model target tracking. Figure 43 shows a screenshot of a live feed, where the model target is being used to detect the object in the real environment. Once the guide view matching successfully locates the object in the live video feed, the guide view locks onto and overlays onto the part in the live video feed and tracks the part as the camera moves around. Once the part has been identified in the video feed, the AR information can then be added to the scene.

Figure 44 shows a screenshot of the demonstration after the object is detected. In this demonstration, individual 3D CAD components were overlayed in color onto the part in the AR environment, as shown in the screenshot. Position error information can then be added to the AR scene next to the components failing inspection. The square nut with the CAD model in red illustrates how this would be done.

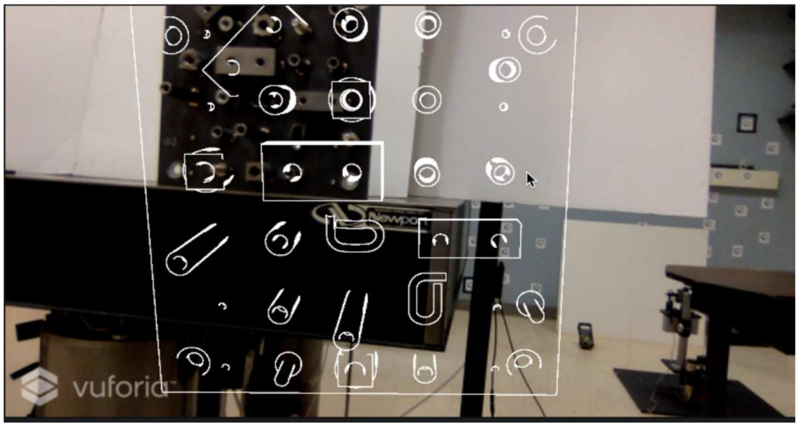


Figure 43. The guide view detects the object in a live video feed.

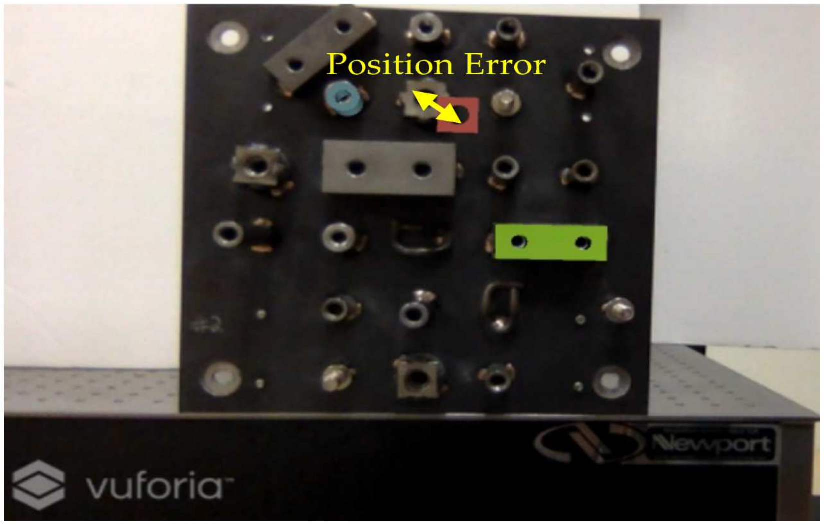


Figure 44. In augmented reality application demonstration, the individual parts are displayed on the screen after object detection, and any potential defects are highlighted.

The AR information can be added to the live video feed once the guide view is successfully located in the live video feed and this depends on the complexity of the guide view – the more complex and unique the guide view, the better. Testing the process using only one component, a block, as the guide view failed because this guide view was not complex and unique enough. Figure 45 shows the block (with insufficient unique features) failing to be detected in the real-world environment.

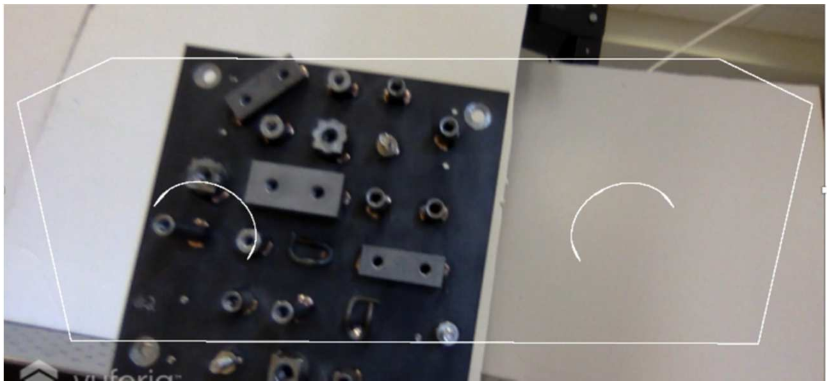


Figure 45. Guide view of the block as seen in the video feed. It cannot detect the block on the plate successfully due to a lack of unique feature identifiers.

This AR application was created in Unity, the cross-engine platform in conjunction with Vuforia, the AR library, to effectively detect the object of interest in a real time video feed, keep tracking it as the object moves and overlay the relevant information on the screen. The model target generator feature of Vuforia library was used to create a guide view to detect our model in the real world. This is where we utilized our CAD model, to create the guide view with the relevant features. Adding photogrammetry with AR provides a both quantitative and visual feedback to the user performing quality inspection.

CHAPTER 6 CONCLUSIONS AND FUTURE WORK

6-1 CONCLUSIONS

In this dissertation, we explored an economical pipeline for large-scale quality inspection in the industrial assembly line environment. The demonstration showed that position errors for components welded onto a steel plate can be estimated at a sub-millimeter level for the artifact studied. An augmented reality (AR) interface then visually displays the position errors for the user on a live video feed of the artifact. The position error measurement is based on acquisition of a 3D point cloud of the assembled system using photogrammetry with images taken with a cell phone and processed with open-source libraries. The position error estimate step was completed with the ICP algorithm for coordinate system definition, coordinate system registration, and finally to estimate individual positional assembly errors of the feature of interest relative to the CAD model. Using a point cloud registration approach like ICP offers more flexibility than object detection (e.g., cylinder detection), allowing us to calculate the position of complex objects and free-form bodies. Augmented reality was effectively employed to create an application to display the relevant information for the user using a smartphone camera or tablet.

Although not as accurate as commercial CMMs or laser scanners, the methodology presented is cost-effective and easily accessible for quality inspection in large-scale assembly application where measurement goals on the scale of $1/10^3$ are adequate. Furthermore, since the only on-site equipment needed is a camera, the quality inspection can be easily performed in areas where CMMs and commercial-grade laser trackers are cost-prohibitive or not practical. The proposed pipeline compromises uncertainty by providing a cost-effective and user-friendly system.

We demonstrated a position error measurement. The exact process can easily be modified to estimate other assembly error measurands like angle deviations. Furthermore, alternating forms of point cloud comparison processes can be explored to study shape defects, particularly object detection, where the geometrical shape of the point cloud is used to align features instead of distance minimization as done in ICP.

The AR component of the project helps the user visualize and locate problem areas. In an industry where assembly quality inspection is often done visually, this pipeline offers a quantitative avenue without compromising the visual aspect for the user.

Uncertainty studies based on random noise on a simulated photogrammetry data set were also performed. Commercial software (PolyWorks) was tested to evaluate the location of a post with added noise through simulation. PolyWorks reported position information for the post with a bias of 0.006 mm for noise as high as 0.5 mm on each point. While this bias is very small compared to assembly tolerances in many applications, the proprietary 'black box' nature of PolyWorks and other commercial software makes it impossible to project general uncertainty limits, to modify the code, and to apply to general complex shapes. The ICP-based approach can be used on general point clouds irrespective of shape and is an open-source algorithm that can be evaluated and modified.

We analyzed three primary contribution sources to the uncertainty in our measurand. First was the noise and possible bias acquired in the 3D reconstruction process. The second is shape differences between the actual components compared to their CAD model, and the third is contributions from data processing choices.

The discussions in Chapter 3 found that the photogrammetry reconstruction process and the shape deviations of the component compared to the CAD model were the limiting factors to our uncertainty studies, limiting the ability to estimate a position error of a 40mm high, 16mm diameter post component to better than a fraction of a millimeter.

Chapter 4 focused on the effect of random noise and point cloud density on the registration process employed to find the position deviation between the two-point clouds. We demonstrated the ideal down sampling ratio of the two-point clouds for our photogrammetry dataset and application. The effect of random noise was studied in detail and at large noise levels it was found that a Rayleigh bias is present, our application was not affected by this bias as the noise levels in the photogrammetry data set were low. In the last chapter an AR application was created to detect and display this positional quality inspection information on a screen, using a cell phone device. The CAD model was used to create a guide view for our application to detect the object in real time and an object tracking class from Vuforia was used to create a footprint of the coordinates as the camera moved.

6-1 FUTURE WORK

A skeletal architecture from start to finish was implemented, and its validity for measurements of millimeter-scale position errors for a sub-meter scale steel assembly was demonstrated. Much of the future work resides in improving the individual steps in the pipeline.

To improve the initial 3D reconstruction, images in different lighting conditions must be explored. Furthermore, the number of images and their orientations have been shown to affect the final 3D reconstruction. A controlled experiment with different quantities of images can be performed to see the impact on noise in the final 3D reconstruction.

Due to worker negligence or specific environmental conditions, best practices in data acquisition might not be possible. One avenue to explore is the use of machine learning to fill in the missing point cloud data using information from CAD.

In the measurand estimation process, we explored the effect of down-sampling the measured point cloud data using a voxel grid and random sampling. Exploring other down-sampling techniques, such as adaptive hierarchical down-sampling [125], which uses a neural network approach, can be performed to customize the down-sampling process according to the 'importance' of each point.

We performed simulations to see the limiting factors on the uncertainty present in the final measurand. Work is still needed to investigate the photogrammetry-based spatial bias in the 3D reconstruction and surface roughness-based deviations. Comparing the photogrammetry reconstructed point cloud with CMM measurements will give a better insight into the reconstruction process's noise contribution. The study analyzed position errors of an 16 mm diameter post with sub-millimeter variations due to the reconstructed point cloud's shape, texture, and noise. More work is required to estimate a comprehensive combined uncertainty, which will depend on the component shape, surface texture, the number and quality of images used for photogrammetry, and photogrammetry reconstruction algorithms for both the sparse and the dense point cloud reconstructions.

Furthermore, in some applications, the position tolerance might call for much lower measurement uncertainty and drive the measurement to the limit where noise in the actual position estimate is on the order of the position error itself. In this limit, a significant bias is present in the measurement result and would need to be removed and a final uncertainty estimated. This limit and the bias was studied in Chapter 4.

The field of AR is ever evolving, and studies can be done to assess the best ways for visual information to be displayed to the user in an AR application. So far, our AR application displays the required data, but further work can be done for repair and maintenance protocols to be issued to the user once a quality defect has been identified.

REFERENCES

1.Babic, M.; Farahani, M.A.; Wuest, T. Image Based Quality Inspection in Smart Manufacturing Systems: A Literature Review. Procedia CIRP 2021, 103, 262–267. https://doi.org/10.1016/j.procir.2021.10.042.

2.Pérez, L.; Rodríguez, Í.; Rodríguez, N.; Usamentiaga, R.; García, D. Robot Guidance Using Machine Vision Techniques in Industrial Environments: A Comparative Review. Sensors 2016, 16, 335. https://doi.org/10.3390/s16030335.

3.Moon, D.; Chung, S.; Kwon, S.; Seo, J.; Shin, J. Comparison and utilization of point cloud generated from photogrammetry and laser scanning: 3D world model for smart heavy equipment planning. Autom. Constr. 2019, 98, 322–331. https://doi.org/10.1016/j.autcon.2018.07.020.

4.Li, J.; Berglund, J.; Auris, F.; Hanna, A.; Vallhagen, J.; Åkesson, K. Evaluation of Photogrammetry for Use in Industrial Production Systems. In Proceedings of the IEEE 14th International Conference on Automation Science and Engineering (CASE), Munich, Germany, 20–24 August 2018; pp. 414–420; https://doi.org/10.1109/COASE.2018.8560496.

5.Sioma, A. 3D imaging methods in quality inspection systems. In Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2019; Proceedings of the Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2019, Wilga, Poland, 27 May–2 June 2019; SPIE: Bellingham, WA, USA, 2019; p. 91; https://doi.org/10.1117/12.2536742

6.Luhmann, T. Close range photogrammetry for industrial applications. ISPRS J. Photogramm. Remote Sens. 2010, 65, 558–569. https://doi.org/10.1016/j.isprsjprs.2010.06.003.

7.Wang, Q.; Kim, M.-K. Applications of 3D point cloud data in the construction industry: A fifteen-year review from 2004 to 2018. Adv. Eng. Inform. 2019, 39, 306–319. https://doi.org/10.1016/j.aei.2019.02.007.

8.González-Jorge, H.; Riveiro, B.; Arias, P.; Armesto, J. Photogrammetry and laser scanner technology applied to length measurements in car testing laboratories. Measurement 2012, 45, 354–363. https://doi.org/10.1016/j.measurement.2011.11.010.

9.Geodetic Systems, Inc. Reports. Available online: https://www.geodetic.com/resources/reports/ (accessed on 31 August 2021).

10.Aldao, E.; González-Jorge, H.; Pérez, J.A. Metrological comparison of LiDAR and photogrammetric systems for deformation monitoring of aerospace parts. Measurement 2021, 174, 109037. https://doi.org/10.1016/j.measurement.2021.109037.

11.Wang, X.; Ong, S.K.; Nee, A.Y.C. A comprehensive survey of augmented reality assembly research. Adv. Manuf. 2016, 4, 1–22. https://doi.org/10.1007/s40436-015-0131-4.

12.de Souza Cardoso, L.F.; Mariano, F.C.M.Q.; Zorzal, E.R. A survey of industrial augmented reality. Comput. Ind. Eng. 2020, 139, 106159. https://doi.org/10.1016/j.cie.2019.106159.

13.Fraunhofer IGD. Quality Checks with Augmented Reality. Avaliable online: https://www.igd.fraunhofer.de/en/press/news/quality-checks-augmented-reality (accessed on 31 August 2021).

14.FARO. Visual Inspect Augmented Reality; FARO: Lake Mary, FL, USA, 2017. Available online: https://www.faro.com/en/Products/Software/Visual-Inspect-Augmented-Reality (accessed on 31 August 2021).

15.PTC. Creo Parametric 3D Modeling Software; PTC: Boston, MA, USA, 2011. Available online: https://www.ptc.com/en/products/creo/parametric (accessed on 19 February 2022).

16.Dai, F.; Lu, M. Photo-Based 3D modeling of construction resources for visualization of operations simulation: Case of modeling a precast façade. In Proceedings of the 2008 Winter Simulation Conference, Miami, FL, USA, 7–10 December 2008; pp. 2439–2446; https://doi.org/10.1109/WSC.2008.4736352.

17.openMVG_main_SfMInit_ImageListing—OpenMVG Library. Available online: https://openmvg.readthedocs.io/en/latest/software/SfM/SfMInit_ImageListing/ (accessed on 18 February 2022).

18.openMVG. OpenMVG (Open Multiple View Geometry). 2021. Available online: https://github.com/openMVG/openMVG (accessed on 31 August 2021).

19.Bianco, S.; Ciocca, G.; Marelli, D. Evaluating the Performance of Structure from Motion Pipelines. J. Imaging 2018, 4, 98. https://doi.org/10.3390/jimaging4080098.

20.Lowe, D.G. Distinctive Image Features from Scale-Invariant Keypoints. Int. J. Comput. Vis. 2004, 60, 91–110. https://doi.org/10.1023/B:VISI.0000029664.99615.94.

21.Cheng, J.; Leng, C.; Wu, J.; Cui, H.; Lu, H. Fast and Accurate Image Matching with Cascade Hashing for 3D Reconstruction. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 1–8; https://doi.org/10.1109/CVPR.2014.8.

22.Moulon, P.; Monasse, P.; Marlet, R. Global Fusion of Relative Motions for Robust, Accurate and Scalable Structure from Motion. In Procedings of the IEEE International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013; pp. 3248–3255; https://doi.org/10.1109/ICCV.2013.403.

23.cDc. OpenMVS: Open Multi-View Stereo Reconstruction Library. 2021. Available online: https://github.com/cdcseacave/openMVS (accessed on 31 August 2021).

24.Ground Control Points Registration—OpenMVG Library. Available online: https://openmvg.readthedocs.io/en/latest/software/ui/SfM/control_points_registration/GCP/ (accessed on 18 February 2022).

25.Register Two Point Clouds Using ICP Algorithm—MATLAB Pcregistericp. Available online: https://www.mathworks.com/help/vision/ref/pcregistericp.html (accessed on 25 September 2021).

26.Li, P.; Wang, R.; Wang, Y.; Tao, W. Evaluation of the ICP Algorithm in 3D Point Cloud Registration. IEEE Access 2020, 8, 68030–68048. https://doi.org/10.1109/ACCESS.2020.2986470.

27.Downsample a 3-D Point Cloud—MATLAB Pcdownsample. Available online: https://www.mathworks.com/help/vision/ref/pcdownsample.html#References (accessed on 18 February 2022).

28.Getting Started with Vuforia Engine in Unity. VuforiaLibrary. Available online: https://library.vuforia.com/getting-started/getting-started-vuforia-engine-unity (accessed on 31 August 2021).

29.Unity Technologies. Unity Real-Time Development Platform. 3D, 2D VR & AR Engine. Available online: https://unity.com/ (accessed on 6 September 2021).

30.Shen, S. Depth-Map merging for Multi-View Stereo with high resolution images. In Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012), Tsukuba, Japan, 11–15 November 2012; pp. 788–791.

31.Girardeau-Montaut, D. (n.d.). CloudCompare - Open Source Project. Retrieved September 8, 2022, from https://www.danielgm.net/cc/

32.Puertas, I., Pérez, C. J. L., Salcedo, D., León, J., Luri, R., & Fuertes, J. P. (2013). Precision study of a coordinate measuring machine using several contact probes. Procedia Engineering, 63, 547–555. https://doi.org/10.1016/j.proeng.2013.08.260

33.Syam, W. P., Rybalcenko, K., Gaio, A., Crabtree, J., &amp; Leach, R. K. (2019). Methodology for the development of in-line optical surface measuring instruments with a case study for additive surface finishing. Optics and Lasers in Engineering, 121, 271–288. https://doi.org/10.1016/j.optlaseng.2019.04.015

34.Tao, F., Qi, Q., Liu, A., &amp; Kusiak, A. (2018). Data-driven smart manufacturing. Journal of Manufacturing Systems, 48, 157–169. https://doi.org/10.1016/j.jmsy.2018.01.006

35.Imkamp, D., Berthold, J., Heizmann, M., Kniel, K., Manske, E., Peterek, M., Schmitt, R., Seidler, J., &amp; Sommer, K.-D. (2016). Challenges and trends in manufacturing measurement technology – the"industrie 4.0" concept. Journal of Sensors and Sensor Systems, 5(2), 325–335. https://doi.org/10.5194/jsss-5-325-2016

36.Zhong, R. Y., Xu, X., Klotz, E., &amp; Newman, S. T. (2017). Intelligent Manufacturing in the context of industry 4.0: A Review. Engineering, 3(5), 616–630. https://doi.org/10.1016/j.eng.2017.05.015

37.Catalucci, S., Thompson, A., Piano, S. et al. Optical metrology for digital manufacturing: a review. Int J Adv Manuf Technol 120, 4271–4290 (2022). https://doi.org/10.1007/s00170-022-09084-5

38.Chen R, Xu J, Zhang S (2020) Digital fringe projection profilometry. In: Leach RK Advances in Optical Form and Coordinate Metrology (IOP Publishing), Chap. 5

39.H. Sjodahl M, Bergstrom P, Fergusson M, Soderholm K, Andersson A (2021) In-line quality control utilizing close-range photogrammetry and a CAD-model. Eng Res Express In Press

40.Bergström P, Fergusson M, Folkesson P, Runnemalm A, Ottosson M, Andersson A, Sjödahl M (2016) Automatic in-line inspection of shape based on photogrammetry. 7th Swedish Prod Symp 1–9

41.Springer London Ltd. (2012). 3D Computer vision.

42.Feng, D., &amp; Feng, M. Q. (2018). Computer vision for SHM of civil infrastructure: From dynamic response measurement to damage detection – a review. Engineering Structures, 156, 105–117. https://doi.org/10.1016/j.engstruct.2017.11.018

43.Flack DR (2001) Good Practice Guide No. 41 CMM measurement strategies (National Physical Laboratory)

44.Pachon, R. M. (2019, March 8). Introduction to photogrammetry. Academia.edu. Retrieved September 9, 2022, from https://www.academia.edu/38516901/Introduction_to_Photogrammetry

45.Moulon, P., Monasse, P., &amp; Marlet, R. (2013). Adaptive structure from motion with a&nbsp;contrario model estimation. Computer Vision – ACCV 2012, 257–270. https://doi.org/10.1007/978-3-642-37447-0_20

46.Moulon, P., Monasse, P., &amp; Marlet, R. (2013). Adaptive structure from motion with a&nbsp;contrario model estimation. Computer Vision – ACCV 2012, 257–270. https://doi.org/10.1007/978-3-642-37447-0_20

47.A block coordinate variable metric forward-backward algorithm. (n.d.). Retrieved September 9, 2022, from https://hal-upec-upem.archives-ouvertes.fr/file/index/docid/945918/filename/BCVMFB_HAL.pdf

48.Moulon, P., Monasse, P., &amp; Marlet, R. (2013, October 15). Global fusion of relative motions for robust, accurate and scalable structure from Motion. Ecole des Ponts ParisTech. Retrieved September 9, 2022, from https://hal-enpc.archives-ouvertes.fr/hal-00873504

49.Moulon, P., Monasse, P., Perrot, R., &amp; Marlet, R. (2017). OpenMVG: Open multiple view geometry. Reproducible Research in Pattern Recognition, 60–74. https://doi.org/10.1007/978-3-319-56414-2_5

50.Agarwal, S., Mierle, K., et al.: Ceres Solver. http://ceres-solver.org

51.Guennebaud, G., Jacob, B., et al.: Eigen v3 (2010). http://eigen.tuxfamily.org

52.Muja, M., Lowe, D.G.: Fast approximate nearest neighbors with automatic algorithm configuration. In: VISAPP (2009)

53.Lowe, D.G.: Distinctive image features from scale-invariant keypoints. IJCV 60(2), 91–110 (2004)

54.Alcantarilla, P.F., Nuevo, J., Bartoli, A.: Fast explicit diffusion for accelerated features in nonlinear scale spaces. BMVC 34(7), 1281–1298 (2013)

55.Rosten, E., Drummond, T.: Machine learning for high-speed corner detection. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 430–443. Springer, Heidelberg (2006). doi:10.1007/11744023

56.Xu, Y., Monasse, P., Géraud, T., Najman, L.: Tree-based morse regions: a topological approach to local feature detection. IEEE Trans. Image Process. 23(12), 5612–5625 (2014)

57.Distinctive image features from scale-invariant keypoints. (n.d.). Retrieved September 9, 2022, from https://www.cs.ubc.ca/~lowe/papers/ijcv04.pdf

58.Tyagi, D. (2020, April 7). Introduction to SIFT(scale invariant feature transform). Medium. Retrieved September 9, 2022, from https://medium.com/data-breach/introduction-to-sift-scale-invariant-feature-transform-65d7f3a72d40

59.L. Pérez, Í. Rodríguez, N. Rodríguez, R. Usamentiaga, and D. García, “Robot Guidance Using Machine Vision Techniques in Industrial Environments: A Comparative Review,” Sensors, vol. 16, no. 3, p. 335, Mar. 2016, doi: 10.3390/s16030335.

60.Why V-stars? Geodetic Systems, Inc. (n.d.). Retrieved September 9, 2022, from https://www.geodetic.com/v-stars/

61.Professional photogrammetry and drone mapping software. Pix4D. (n.d.). Retrieved September 9, 2022, from https://www.pix4d.com/

62.Drone mapping software: Drone mapping app: UAV mapping: Surveying software. DroneDeploy. (n.d.). Retrieved September 9, 2022, from https://www.dronedeploy.com/

63.Contextcapture: 3D reality modeling software. Bentley. (n.d.). Retrieved September 9, 2022, from https://www.bentley.com/en/products/brands/contextcapture

64.RealityScan. RealityCapture - 3D Models from Photos and/or Laser Scans. (n.d.). Retrieved September 9, 2022, from https://www.capturingreality.com/

65.Cdcseacave. (n.d.). Home · cdcseacave/openmvs wiki. GitHub. Retrieved September 9, 2022, from https://github.com/cdcseacave/openMVS/wiki

66.S. Shen, "Depth-map merging for Multi-View Stereo with high resolution images," Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012), 2012, pp. 788-791

67.Blahnik, V., &amp; Schindelbeck, O. (2021, June 1). Smartphone imaging technology and its applications. De Gruyter. Retrieved September 9, 2022, from https://www.degruyter.com/document/doi/10.1515/aot-2021-0023/html?lang=en

68.K. Matsumoto, T. Nakamura, A. Yusa and S. Nagai, "A new MOS phototransistor operating in a non-destructive readout mode," Jpn. J. Appl. Phys., vol. 24, no. 5A, p. L323, 1985, https://doi.org/10.1143/JJAP.24.L323.Search in Google Scholar

69.American Society for Photogrammetry and Remote Sensing (2019). What is ASPRS?. Available online at: https://www.asprs.org/organization/what-is-asprs.html (accessed November 27, 2019)Google Scholar

70.Dolgui, A., Sgarbossa, F., &amp; Simonetto, M. (2021). Design and Management of Assembly Systems 4.0: Systematic Literature Review and Research Agenda. International Journal of Production Research, 60(1), 184–210. https://doi.org/10.1080/00207543.2021.1990433

71.How to benefit from three different types of quality inspections. NGC Software. (n.d.). Retrieved September 9, 2022, from https://www.ngcsoftware.com/post/140/how-to-benefit-from-three-different-types-of-quality-inspections/

72.Poverty and education; education policy series; vol.:10; 2008. (n.d.). Retrieved September 9, 2022, from https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.464.9607&amp;rep=rep1&amp;type=pdf

73.Proceedings Volume 1611, Sensor Fusion IV: Control Paradigms and Data Structures; (1992) https://doi.org/10.1117/12.57955

74.CSDL: IEEE Computer Society. CSDL | IEEE Computer Society. (n.d.). Retrieved September 9, 2022, from https://www.computer.org/csdl/proceedings-article/robot/1989/00099983/12OmNx6g6dy

75.1 fast and robust iterative closest point - arxiv. (n.d.). Retrieved September 9, 2022, from https://arxiv.org/pdf/2007.07627.pdf

76.S. Rusinkiewicz and M. Levoy, "Efficient variants of the ICP algorithm," Proceedings Third International Conference on 3-D Digital Imaging and Modeling, 2001, pp. 145-152, doi: 10.1109/IM.2001.924423.

77.Registration of Laser Scanning Point Clouds: A review - researchgate. (n.d.). Retrieved September 9, 2022, from https://www.researchgate.net/publication/325293020_Registration_of_Laser_Scanning_Point_Clouds_A_Review

78.Point feature histograms (PFH) descriptors. Point Feature Histograms (PFH) descriptors - Point Cloud Library 0.0 documentation. (n.d.). Retrieved September 9, 2022, from https://pcl.readthedocs.io/projects/tutorials/en/latest/pfh_estimation.html#pfh-estimation

79.R. B. Rusu, N. Blodow and M. Beetz, "Fast Point Feature Histograms (FPFH) for 3D registration," 2009 IEEE International Conference on Robotics and Automation, 2009, pp. 3212-3217, doi: 10.1109/ROBOT.2009.5152473.

80.Nikolov, I., &amp; Madsen, C. (2020, October 8). Rough or noisy? metrics for noise estimation in SFM reconstructions. Sensors (Basel, Switzerland). Retrieved September 9, 2022, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7582591/

81.Bentley ContextCapture. [(accessed on 20 September 2019)];2016 Available online: https://www.bentley.com/

82.Agisoft Metashape. [(accessed on 20 September 2019)];2010 Available online: http://www.agisoft.com/

83.Nealen, A., Igarashi, T., Sorkine, O., &amp; Alexa, M. (2015, August 20). Laplacian mesh optimization. NYU Scholars. Retrieved September 9, 2022, from https://nyuscholars.nyu.edu/en/publications/laplacian-mesh-optimization

84.Lu X., Chen W., Schaefer S. Robust mesh denoising via vertex pre-filtering and l1-median normal filtering. Comput. Aided Geom. Des. 2017;54:49–60. doi: 10.1016/j.cagd.2017.02.011

85.Lu X., Liu X., Deng Z., Chen W. An efficient approach for feature-preserving mesh denoising. Opt. Lasers Eng. 2017;90:186–195. doi: 10.1016/j.optlaseng.2016.09.003.

86.3D reconstruction scene error analysis - ABCM. (n.d.). Retrieved September 9, 2022, from https://www.abcm.org.br/upload/files/PI_II_08.pdf

87.PolyWorks 3d metrology software. Engineering &amp; Manufacturing Services. (2019, February 10). Retrieved September 9, 2022, from https://www.ems-usa.com/products/3d-software/polyworks/

88.Bottani, E., &amp; Vignali, G. (2019). Augmented Reality Technology in the manufacturing industry: A review of the last decade. IISE Transactions, 51(3), 284–310. https://doi.org/10.1080/24725854.2018.1493244

89.Tamura, Hideyuki. “Steady Steps and Giant Leap Toward Practical Mixed Reality Systems and Applications.” (2002).

90.Ivan Sutherland and Bob Sproull create the first virtual reality head mounted display system. Ivan Sutherland and Bob Sproull Create the First Virtual Reality Head Mounted Display System : History of Information. (n.d.). Retrieved September 9, 2022, from https://www.historyofinformation.com/detail.php?entryid=1087

91.Van Krevelen, D. W. F., &amp; Poelman, R. (2010). A survey of Augmented Reality Technologies, applications and limitations. International Journal of Virtual Reality, 9(2), 1–20. https://doi.org/10.20870/ijvr.2010.9.2.2767

92.Van Krevelen, D.W.F. and Poelman, R. (2010) A survey of augmented reality technologies, applications and limitations. The International Journal of Virtual Reality, 9(2), 1–20

93.Bridging multiple user interface dimensions with augmented reality. (n.d.). Retrieved September 9, 2022, from https://arbook.icg.tugraz.at/schmalstieg/Schmalstieg_039.pdf

94.P. Fite-Georgel, "Is there a reality in Industrial Augmented Reality?," 2011 10th IEEE International Symposium on Mixed and Augmented Reality, 2011, pp. 201-210, doi: 10.1109/ISMAR.2011.6092387

95.Vignali, G., Bertolini, M., Bottani, E., Di Donato, L., Ferraro, A., &amp; Longo, F. (2017). Design and testing of an augmented reality solution to enhance operator safety in the Food Industry. International Journal of Food Engineering, 14(2). https://doi.org/10.1515/ijfe-2017-0122

96.Segovia, D., Mendoza, M., Mendoza, E., &amp; González, E. (2015). Augmented reality as a tool for production and quality monitoring. Procedia Computer Science, 75, 291–300. https://doi.org/10.1016/j.procs.2015.12.250

97.Azuma, R. T. (1997). A survey of Augmented Reality. Presence: Teleoperators and Virtual Environments, 6(4), 355–385. https://doi.org/10.1162/pres.1997.6.4.355

98.Portalés C, Casas S, Gimeno J, Fernández M, Poza M. From the Paper to the Tablet: On the Design of an AR-Based Tool for the Inspection of Pre-Fab Buildings. Preliminary Results of the SIRAE Project. Sensors. 2018; 18(4):1262. https://doi.org/10.3390/s18041262

99.Zhang, Z. (2014). Camera parameters (intrinsic, extrinsic). Computer Vision, 81–85. https://doi.org/10.1007/978-0-387-31439-6_152

100.Monochrome vs color sensors - opto engineering basics. (n.d.). Retrieved September 9, 2022, from https://www.opto-e.com/basics/monochrome-vs-color-sensors

101.Zhang, M. (2021, February 4). Apple invents a camera with 3 sensors and a prism that splits light. PetaPixel. Retrieved September 9, 2022, from https://petapixel.com/2015/03/26/apple-invents-a-camera-with-3-sensors-and-a-prism-that-splits-light/

102.Digital orthophoto. Orthophoto, Digital Orthophoto, Ortho-Rectified. (n.d.). Retrieved September 9, 2022, from http://www.photomapping.com.au/digital-orthophoto

103.Basics of photogrammetry. GIS Resources. (2020, September 20). Retrieved September 9, 2022, from https://gisresources.com/basic-of-photogrammetry_2/

104.PhotoModeler. (n.d.). Retrieved September 9, 2022, from https://www.photomodeler.com/how-does-photogrammetry-work/

105.Assessing the performance of a structured light scanner - isprs.org. (n.d.). Retrieved September 9, 2022, from https://www.isprs.org/proceedings/XXXVIII/part5/papers/177.pdf

106.Poverty and education; education policy series; vol.:10; 2008. (n.d.). Retrieved September 9, 2022, from https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.464.9607&amp;rep=rep1&amp;type=pdf

107.Photogrammetry and structured light: comparison and integration of techniques in survey of the Corsini Throne at Corsini Gallery in RomeRetrieved September 9, 2022, from https://www.imeko.info/publications/tc4-Archaeo-2019/IMEKO-TC4-METROARCHAEO-2019-32.pdf

108.High-resolution 3D digital models of artworks - researchgate.net. (n.d.). Retrieved September 9, 2022, from https://www.researchgate.net/profile/Roberto-Scopigno/publication/252289473_High-resolution_3D_digital_models_of_artworks/links/00b7d53a1a13a78228000000/High-resolution-3D-digital-models-of-artworks.pdf

109.Optica Publishing Group. (n.d.). Retrieved September 9, 2022, from https://opg.optica.org/aop/fulltext.cfm?uri=aop-3-2-128&amp;id=211561

110.By. (2015, January 14). Laser Scanning vs. photogrammetry. Lanmar Services. Retrieved September 9, 2022, from http://lanmarservices.com/2014/11/07/laser-scanning-vs-photogrammetry/

111.Disjoint–set data structure (union–find algorithm). Techie Delight. (2022, May 31). Retrieved September 9, 2022, from https://www.techiedelight.com/disjoint-set-data-structure-union-find-algorithm/

112.Brute force matching between camera shots and synthetic images from ... (n.d.). Retrieved September 9, 2022, from https://www.researchgate.net/profile/Richard-Boerner/publication/307529413_brute_force_matching_between_camera_shots_and_synthetic_images_from_point_clouds/links/586e82c008ae6eb871be0e3f/brute-force-matching-between-camera-shots-and-synthetic-images-from-point-clouds.pdf

113.Feature matching. OpenCV. (n.d.). Retrieved September 9, 2022, from https://docs.opencv.org/3.4/dc/dc3/tutorial_py_matcher.html

114.Hindawi. (n.d.). Grasp detection under occlusions using SIFT features. Grasp Detection under Occlusions Using SIFT Features. Retrieved September 9, 2022, from https://www.hindawi.com/journals/complexity/2021/7619794/

115.Introduction to homography. OpenCV. (n.d.). Retrieved September 9, 2022, from https://docs.opencv.org/4.x/d9/dab/tutorial_homography.html

116.Pyakurel, A. (2020, August 19). Understanding the dense point-cloud generation (DensifyPointCloud) in openmvs. Medium. Retrieved September 9, 2022, from https://aashutosh-py.medium.com/understanding-the-dense-point-cloud-generation-densifypointcloud-in-openmvs-faa6777c7501

117.What is Industry 4.0 and how does it work? IBM. (n.d.). Retrieved September 9, 2022, from https://www.ibm.com/topics/industry-4-0

118.6.3DdataProcessing. Welcome &lt; 3D Vision Laboratory. (n.d.). Retrieved September 9, 2022, from https://www.cvl.iis.u-tokyo.ac.jp/class2016/2016w/papers/6.3DdataProcessing/

119.Understanding computation time - research-repository.st-andrews.ac.uk. (n.d.). Retrieved September 9, 2022, from https://research-repository.st-andrews.ac.uk/bitstream/handle/10023/24947/Computation_Time_paper.pdf

120.Ptcloudin. Fit plane to 3-D point cloud - MATLAB. (n.d.). Retrieved September 9, 2022, from https://www.mathworks.com/help/vision/ref/pcfitplane.html.

121.Morse, Ed, and Herb Voelcker. "A Tale of Two Tails." *Brown & Sharpe Publication of Precision Manufacturing*.

122."Gom Inspect pro: Making Quality Easily Visible: Software for 3D Inspections." *Making Quality Easily Visible | Software for 3D Inspections*, https://www.gom.com/en/products/gom-suite/gom-inspect-pro.

123."Distances Computation." *Distances Computation - CloudCompareWiki*, https://www.cloudcompare.org/doc/wiki/index.php/Distances_Computation.

124."To-Cloud Distance." *Cloud*, https://www.cloudcompare.org/doc/wiki/index.php/Cloud-to-Cloud_Distance.

125.Nezhadarya, Ehsan, et al. “Adaptive Hierarchical down-Sampling for Point Cloud Classification.” *CVF Open Access*, 1 Jan. 1970, https://openaccess.thecvf.com/content_CVPR_2020/html/Nezhadarya_Adaptive_Hierarchical_Down-Sampling_for_Point_Cloud_Classification_CVPR_2020_paper.html.

126.Nawab, Rahma, and Angela Davies Allen. “Low-Cost Ar-Based Dimensional Metrology for Assembly.” *Machines*, vol. 10, no. 4, 2022, p. 243., https://doi.org/10.3390/machines10040243.

127.Rieke-Zapp, D., et al. “Evaluation of the Geometric Stability and the Accuracy Potential of Digital Cameras - Comparing Mechanical Stabilisation versus Parameterisation.” *ISPRS Journal of Photogrammetry and Remote Sensing*, Elsevier, 11 Nov. 2008.

128.Alenyà, G., et al. “TOF Cameras for Active Vision in Robotics.” *Sensors and Actuators A: Physical*, vol. 218, 2014, pp. 10–22., https://doi.org/10.1016/j.sna.2014.07.014.