



CHALMERS
UNIVERSITY OF TECHNOLOGY



Defect-aware 3D Geometry Reconstruction from 2D Annotations and 3D Scan Data

Master's Thesis in Product Development

ABHISHEK HALWAI

DURVESH PONKSHE

DEPARTMENT OF INDUSTRIAL AND MATERIALS SCIENCE

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2026
www.chalmers.se

MASTER'S THESIS 2026

Defect-aware 3D Geometry Reconstruction from 2D Annotations and 3D Scan Data

Master's Thesis in Product Development

ABHISHEK HALWAI
DURVESH PONKSHE



CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Industrial and Materials Science
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2026

Defect-aware 3D Geometry Reconstruction from 2D Annotations and 3D Scan Data
Master's Thesis in Product Development
Abhishek Halwai
Durvesh Ponkshe

© ABHISHEK HALWAI, DURVESH PONKSHE 2026.

Supervisor: Roham Sadeghi Tabar, Department of Industrial and Materials Science
Examiner: Roham Sadeghi Tabar, Department of Industrial and Materials Science

Master's Thesis 2026
Department of Industrial and Materials Science
Division of Product Development
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in L^AT_EX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2026

Abstract

Components produced through additive manufacturing require particular methods of inspection to locate and characterize surface defects on those components. Although 2D image-based defect detection is an accepted method, linking that 2D defect information to the 3D geometry of the part by reconstructing the location and severity of the defect-induced deformation of the 3D surface is challenging with limited 3D scan data and the absence of certified 3D reference geometry. This gap between 2D defect annotation and 3D defect geometry motivates this thesis. This master’s thesis develops and analyzes a pipeline that projects 2D defect annotations onto a nominal 3D CAD model to reconstruct defect-aware 3D geometry with the help of structured-light scan data from the 3D-ADAM dataset. The pipeline projects 2D defect masks to 3D nominal surface points through the sensor’s organized point cloud, registers each PLY scan to the nominal CAD, computes the signed deviation between defective and nominal scans, and finally reconstructs the defect on the CAD model in the form of a point cloud, a mesh, and a deviation map. The thesis also explores the possibility of predicting defect deviation directly on the nominal CAD using a PointNet++ model. A geometry-only formulation shows significant limitations in predicting deviation magnitude and sign as this information is unavailable in the nominal geometry. After additionally providing a coarse 2D heatmap of the deviation as input, the magnitude prediction error was reduced by around 28% relative to the geometry-only model and 36% relative to the class-average baseline, and the direction of deviation was also recovered. The accuracy of defect placement onto the 3D surface is bounded by single-view registration and partially unresolved mirror ambiguity. This thesis validates the scan-based defect mapping and reconstruction pipeline, and describes the information required to predict defect deviation. Furthermore, this thesis identifies RGB-based defect prediction to reduce reliance on the 3D scan at inference as the key direction for future research.

Keywords: 3D reconstruction, defective geometry, geometric deep learning, point cloud, PointNet++.

Acknowledgements

First and foremost, we would like to express our gratitude to our supervisor and examiner, Roham Sadeghi Tabar, for providing us with the thesis topic and the opportunity to conduct our master's thesis, which is a requirement for completing our M.Sc. degree in Product Development. His valuable feedback and constant support have played an essential role in shaping and aligning the thesis in the correct direction.

Furthermore, we would like to extend our appreciation to the Department of Industrial and Materials Science for providing valuable resources to help us progress our master's thesis.

Lastly, we would like to thank everyone else who has not been specifically mentioned but has contributed to the completion of this master's thesis.

Abhishek Halwai and Durvesh Ponkshe
Gothenburg, September 2026

List of Acronyms

The acronyms used throughout this thesis are listed below in alphabetical order:

AM	Additive Manufacturing
AUROC	Area Under the Receiver Operating Characteristic
CAD	computer-aided design
CFA	Coupled hypersphere-based Feature Adaptation
CNN	Convolutional neural network
CT	Computed tomography
FDM	Fused deposition modeling
HPC	High-Performance Computing
ICP	Iterative Closest Point
IoU	Intersection over Union
M3DM	Multi-3D-Memory
MAE	Mean Absolute Error
MLP	Multi-layer perceptron
NDT	Non-destructive testing
NLP	Natural Language Processing
NPY	NumPy Array
NPZ	NumPy Zipped
PaDiM	Patch Distribution modeling
PCA	Principal Component Analysis
PLY	Polygon File Format
QA	Quality assurance
RGB	Red, Green, and Blue
RNN	Recurrent neural network
SAM	Segment Anything Model
STL	Standard Triangle Language
VTK	Visualization Toolkit

Nomenclature

Variables

Below is the list of variable symbols and their meanings used throughout the thesis, listed in alphabetical order:

Symbol	Meaning
A	surface area of the defect (mm^2)
A_{part}	surface area of nominal part (mm^2)
A_t	area of triangle t
$b(p)$	local baseline: median deviation of the surrounding nominal surface points at point p
$\bar{d}$	(mean depth) mean absolute deviation over the defect region
$d(p)$	raw measured deviation at point p
$d'(p)$	corrected deviation at point p after high-pass filtering
d_i	signed deviation at region point i
d_{max}	deviation clamp: maximum retained deviation magnitude
G	organized point cloud grid
H	grid height (1024)
k	number of nearest neighbors for the local baseline (around 50)
N_r	number of defect region points
N_{total}	total number of sampled points
P	a 3D point (x, y, z)
r_{reg}	region radius for projecting a defect point onto the CAD (3 mm)
V	volume: net material change (mm^3)
W	grid width (1280)
W_m	defect mask crop width (1024)
z_{med}	median depth of a point set
δ	column offset (128)
τ_{fit}	pose-fit tolerance (2 mm)
τ_z	depth-clean band (0.045 m)

Contents

List of Acronyms	ix
Nomenclature	xi
List of Figures	xv
List of Tables	xvii
1 Introduction	1
1.1 Background	1
1.2 Purpose	2
1.2.1 Research Questions	2
1.3 Scope	2
1.3.1 Hypotheses	3
2 Literature Study	5
2.1 Manufacturing Industry and Quality Assurance Standards	5
2.2 Dataset	6
2.3 Anomaly Detection	7
2.4 Part Segmentation	8
2.5 Registration	8
2.5.1 Point Cloud Registration via Iterative Closest Point Algorithm	9
2.5.1.1 PCA-based Initial Alignment and Scaling	9
2.5.1.2 Reference Plane Alignment	9
2.5.1.3 Registration Quality Metric	10
2.6 PointNet and PointNet++	10
2.7 Learning-Based Geometry Reconstruction	11
2.8 The Research Gap	12
3 Methodology	15
3.1 Research Methodology	15
3.2 Overall Pipeline Architecture	15
3.3 Data Preparation	16
3.4 Anomaly Detection and Part Segmentation	18
3.4.1 Anomaly Detection	19
3.4.2 Part Segmentation using SAM	19
3.4.3 Use of Dataset Annotations	19

3.5	Defect-to-Surface Mapping	20
3.5.1	Mapping	20
3.5.2	Verification	21
3.6	Pose Estimation and Signed Deviation Computation	22
3.6.1	Pose Estimation	23
3.6.2	Signed Deviation	25
3.6.3	Coarse Heatmap Generation	26
3.6.4	Quality Filtering	27
3.7	Deviation Prediction	28
3.8	Geometry Reconstruction	30
3.9	Defect Quantification	31
4	Results	33
4.1	Anomaly Detection and Part Segmentation	33
4.1.1	Anomaly Detection	33
4.1.2	SAM Part Segmentation	33
4.2	Defect-to-Surface Mapping	35
4.3	Pose Estimation / Alignment	36
4.4	Deviation Prediction	37
4.4.1	Magnitude	38
4.4.2	Sign Accuracy	38
4.4.3	Interpretation	40
4.5	Geometry Reconstruction	40
4.5.1	Reconstruction for Each Defect Type	42
4.5.2	Measured vs Predicted	46
4.5.3	Mirror Ambiguity and Fidelity Exception	47
4.5.4	Summary of Geometry Reconstruction Results	49
4.6	Defect Quantification	49
5	Discussion	51
5.1	Interpretation of the Results	51
5.2	The Scan-free Question	51
5.3	Design Choices and Trade-offs	52
5.4	Limitations	53
5.4.1	Limitations Mitigated	53
5.4.2	Limitations Dictating Thesis Work	53
5.4.3	Limitations Out of Scope	53
5.5	Future Work	54
6	Conclusion	55
	Bibliography	56
A	Appendix 1	I
A.1	Deviation Prediction Results	I
A.2	Geometry Reconstruction Results	I
A.3	Quantification Results	II

List of Figures

3.1	Design Research Methodology Framework	16
3.2	Proposed Pipeline Architecture in the form of function blocks for each phase	17
3.3	RGB images of all three instances of sample 006 of part 3M2C with nominal condition from 3D ADAM dataset [19]	18
3.4	Sample 006 of part 3M2C (a) Ground-Truth annotation of Bulge defect (b) Binary Part Mask	20
3.5	Architecture of 2D-to-3D Mapping stage	21
3.6	Part 3M2C (a) Nominal STL Geometry (b) Point Sampled Geometry	23
3.7	Three Initial Pose Estimation Strategies used for Registering 3D Scan to CAD	24
3.8	Manual Correspondence attempt for pose estimation by choosing 3D points on (a) PLY Scan from dataset (b) nominal CAD	25
3.9	Coarse 2D Heatmap of part 3M2C (a) Sample 001 with Under-Extrusion (material removing) defect (b) Sample 014 with Over-Extrusion (material adding) defect	27
3.10	Architecture of PointNet++ network	29
4.1	SAM segmentation representation for 3M2C with three viewpoints: RGB image, binary SAM mask, Overlay with mask boundary on RGB image	34
4.2	2D-to-3D Mapping verification with on-part fraction and valid fraction	35
4.3	Test MAE at every Epoch for Heatmap model	37
4.4	Training and Validation Loss plot across all epochs of Heatmap Model	38
4.5	Deviation magnitude prediction comparison of both models with baseline	39
4.6	Deviation prediction sign accuracy comparison of both models with baseline	39
4.7	Output forms of Reconstruction of 006 sample of 3M2C part with Bulge defect in (a) Deviated Point Cloud (b) VTK Deviation Color Map along with (c) Overlay image	41
4.8	Derived Output forms of Reconstruction of 006 sample of 3M2C part with Bulge defect in (a) Screened Poisson Surface Reconstruction (b) Exported STL after Surface Reconstruction	41
4.9	Deviated Point Cloud of sample 009 of part 1M2 with Crack defect	42

4.10	Sample 009 of part 1M2 with Crack Defect Reference from Dataset (a) RGB Image (b) Ground-Truth Annotation	43
4.11	Deviated Point Cloud of sample 011 of part 3M2C with Cut defect . .	43
4.12	Sample 011 of part 3M2C with Cut Defect Reference from Dataset (a) RGB Image (b) Ground-Truth Annotation	43
4.13	Sample 004 of part 4M2 with Hole Defect (a) Deviated Point Cloud (b) Overlay Image	44
4.14	Sample 014 of part 3M2C with Over-Extrusion Defect (a) Deviated Point Cloud (b) Overlay Image	44
4.15	Sample 001 of part 3M2C with Under-Extrusion Defect (a) Deviated Point Cloud (b) Overlay Image	45
4.16	Deviated Point Cloud of sample 006 of part 3M2C with bulge defect with (a) Measured Deviation (b) Predicted Deviation	46
4.17	Mirror Ambiguity in sample 002 of part 4M2 with Under-Extrusion defect (a) Deviated Point Cloud (b) Overlay image	48
4.18	Fidelity Exception in sample 014 of part 2M2M with Bulge defect (a) Deviated Point Cloud (b) Overlay image	48
4.19	Mean depth prediction error for each defect type by heatmap-enabled model	50
A.1	Zoomed view of crack sample 009 for part 1M2	II
A.2	Zoomed view of cut sample 011 for part 3M2C	II
A.3	Zoomed view of hole sample 004 for part 4M2	IV
A.4	Zoomed view of over-extrusion sample 014 for part 3M2C	IV
A.5	Zoomed view of under-extrusion sample 001 for part 3M2C	IV

List of Tables

2.1	Current thesis compared with related learning-based approaches . . .	12
4.1	Image-level AUROC comparison across five anomaly detection models on part 3M2C.	33
4.2	Per-defect image-level AUROC (%) for each model on part 3M2C. . .	34
4.3	Mean per-part pose-estimation metrics for accepted samples	36
4.4	Prediction Error and Standard Deviation per Defect for Heatmap- enabled Model	47
A.1	Comparison of magnitude MAE (mm) per defect between baseline, geometry-only model and heatmap-enabled model at epoch 80	I
A.2	Comparison of Sign Accuracy on significant points between geometry- only model and heatmap-enabled model at epoch 80	I
A.3	Three quantification measures by part and defect type	III

1

Introduction

In manufacturing, manufactured parts often differ from their intended design because of surface defects and extrusion errors. The identification and characterization of these defects in three dimensions is important for quality control and rectifying the manufacturing process. The current inspection methods either depend on manual evaluation or produce two-dimensional defect information which is not directly relevant to the 3D part geometry, which creates a gap between the defect location in the image and its actual location on the part. This thesis addresses that gap by creating a pipeline that maps 2D defect annotations onto the 3D nominal CAD geometry and reconstructs defect-aware 3D models with the help of structured-light scan data. The industrial motivation is large-scale automated characterization of 3D defects, while the scientific question is what information is required for correlating 2D defect data to 3D geometry and to what extent it is possible.

1.1 Background

In manufacturing and repair-oriented industrial production settings, parts arrive with highly variable and unknown conditions such as wear, scratches, dents, missing material, or surface contamination. Today, assessment of such parts is carried out either through manual visual inspection or by using 3D scanning equipment, both of which are time-consuming, operator-dependent, and difficult to scale across large part families. This results in a bottleneck for industrial repair and re-manufacturing, where fast and consistent decisions are essential. Recent advances in computer vision and deep learning fields have enabled automatic defect detection and segmentation from images. By combining these techniques with existing nominal geometry, it becomes possible to reconstruct the 3D shape of defects by estimating defect depth and shape. Instead of relying on a 3D scan of every damaged part, images from industrial cameras, along with the known nominal part shape, can be used to infer the geometry of defects on the component surface. In this thesis, publicly available datasets such as 3D-ADAM with industrial sensor data and defect annotations are used to benchmark defect localization and develop such image-driven methods. From a scalable manufacturing and re-manufacturing perspective, there is a clear need in industry for a fast and reliable method to detect and localize defects on complex parts from inspection images rather than relying on 3D scans. Such a method would regenerate a digital model containing both the original geometry and the defect geometry, thereby explicitly capturing the as-is state of a part. This model is useful for downstream steps such as repair planning, machining allowance definition,

tolerance/fit analysis and digital twin updates. This need motivates research into image-based methods that can reconstruct defect geometry on top of nominal CAD without requiring 3D scanning at inference time.

1.2 Purpose

The purpose of this thesis is to investigate the feasibility of a workflow that detects, localizes and reconstructs defect geometry on a nominal CAD model represented by triangulated STL geometry. This thesis also aims to determine the information required for scan-free defect prediction during inference. By estimating defect shape and depth from inspection data and nominal geometry, the thesis aims to enable the creation of defect-aware 3D digital models that can support manufacturing and re-manufacturing applications.

1.2.1 Research Questions

1. To what extent can a pipeline map 2D defect annotations onto a 3D nominal CAD model and reconstruct defect-aware 3D geometry, using structured-light scan data to build the mapping and measure deviation?
2. How well can defect deviation magnitude and its direction be predicted on the nominal CAD, and what input information is required for scan-free inference?

1.3 Scope

This thesis investigates the mapping of 2D defect annotations onto 3D nominal CAD geometry and reconstruction of defect-aware 3D models, assessed on parts in the 3D-ADAM dataset that have corresponding CAD models. Automated anomaly detection and segmentation were explored but scoped out as pre-processing rather than a core thesis contribution; the pipeline instead used the ground-truth defect and part annotations (binary masks) from the dataset in order to focus on the 2D-to-3D mapping, deviation measurement and defective geometry reconstruction.

The pipeline is developed and validated using structured-light scan data from the dataset. A complete scan-free approach is not possible and is not claimed because the mapping of defective 3D scans to their corresponding 2D images is required to build and train the model for prediction. Thus, this approach enables defect prediction during inference without any new 3D scan as the trained model can predict the deviation for similar defects on similar parts based only on the mapped 2D information. Hence, the 3D scans are required to create the prediction model and not for inspection of each subsequent part. This concept of scan-based training with scan-free inference for similar defects is investigated in this thesis as a research direction rather than as a capability delivered by this thesis. The quantitative defect descriptors such as area, depth, and volume are documented as approximate indicators of defect severity, and not as certified measurements. Parts without CAD models, as well as digital twin integration, are outside the scope.

1.3.1 Hypotheses

This thesis is based on the following hypotheses:

- The nominal CAD geometry of each part in the dataset is available and can be used as a fixed reference on which defects are expressed.
- The sensor's structured point cloud enables 2D defect annotations to be mapped onto 3D surface points based on a one-to-one correspondence without any additional reconstruction step for this 3D-ADAM dataset.
- With adequate training data, a geometric learning model can predict defect deviation on the nominal CAD geometry from the provided defect region alone, allowing scan-free inference.

2

Literature Study

2.1 Manufacturing Industry and Quality Assurance Standards

Manufacturing quality assurance (QA) has historically depended on manual visual inspection and non-destructive testing (NDT) techniques. Although flexible and widely adopted, manual inspection is highly operator-dependent, inconsistent across shift changes, and unable to maintain the throughput demanded by modern high-volume production lines [5]. In additive manufacturing specifically, which is the process family this thesis addresses, common defects include bulging, under-extrusion, over-extrusion, surface holes, and cuts, each of which arises from different process disturbances such as inconsistent material flow, thermal gradients, or tool path errors [5]. Other manufacturing processes exhibit different characteristic defect types such as porosity and shrinkage in casting, or burrs and residual stress in machining, but these fall outside of the thesis scope, which focuses on the additive manufacturing parts provided in the 3D-ADAM dataset (Section 3.3). These defects can compromise the structural integrity and dimensional accuracy of the finished part, with consequences ranging from functional failure to costly product recalls [25]. Human inspectors are susceptible to fatigue errors, and the subjective nature of visual assessment makes it difficult to standardize defect classification [27]. In addition, the surface-defect detectability depends on characteristics such as defect size, contrast, and boundary clarity, making small or low-contrast anomalies particularly challenging to identify reliably [20].

Conventional NDT techniques, which include ultrasonic testing, eddy-current inspection, and X-ray computed tomography (CT), all address some of the limitations by providing quantitative, repeatable measurements [5]. 3D laser scanning and structured light scanning can capture high-resolution surface geometry by producing point clouds or mesh representations, which encode surface deviations relative to a nominal CAD model. These methods, however, involve trade-offs, including longer acquisition, preprocessing, and registration times before inspection, often requiring skilled operators, and ultimately have long cycle times that are incompatible with in-line inspection [1].

Although 3D laser and structured-light scanning provide accurate surface geometry, they require an additional measurement and processing step. In geometry assurance applications, individual manufactured components may be scanned and the resulting data stored for subsequent geometric analysis [26]. This creates a need for geometry-specific reference data that pair 2D images with 3D scans of the same

part, which is discussed next.

2.2 Dataset

Evaluation of the currently available datasets was necessary before selecting a dataset for this thesis. The 3D-ADAM dataset was considered a prospective dataset, but it was first evaluated against benchmark datasets which contributed to the field of anomaly detection in the last few years and also influenced the creation of the 3D-ADAM dataset: MVTec AD, MVTec 3D-AD, and Real3D-AD. MVTec AD is a comprehensive 2D industrial anomaly detection dataset featuring high-resolution RGB images along with pixel-wise defect masks, but it supports only image-based analysis and does not provide depth information, point clouds, or machine-element annotations for mapping of defects to geometry. Although it offers a broad set of surface anomalies, this dataset was captured under controlled conditions that do not reflect the real manufacturing environment variations, such as varying viewpoint, multiple sensors, lighting variations, and partial occlusions [2]. MVTec 3D-AD further expands the MVTec idea into 3D by including high-resolution scans useful for 3D unsupervised anomaly detection and localization; however, it remains limited to 10 object categories, which are rather limited compared to the industrial anomaly detection tasks [3]. The Real3D-AD dataset advances the field by providing high-precision point cloud scans of 12 industrial prototype categories and by benchmarking current anomaly detection methods using point clouds; however, it remains limited in geometric diversity and does not provide multimodal RGB and 3D pairing, multiple sensor acquisition, or machine-element annotations [16].

In contrast, 3D-ADAM addresses these limitations by featuring a more advanced and realistic benchmark for industrial additive manufacturing inspection. 3D-ADAM includes 14,120 high-resolution RGB images and 3D scans of 217 unique parts, captured using four industrial sensors, together with 27,346 defect annotations across 12 defect classes and 8,110 machine element annotations across 16 classes. Compared to earlier datasets, 3D-ADAM considers additional practical inspection challenges such as sensor viewpoint, part pose variations, illumination, and partial occlusion, and it also provides synchronized pairs of RGB images and point clouds supporting both anomaly localization and reconstruction of geometry. Thus, 3D-ADAM is more suitable for the scope of this thesis since it provides the 2D anomalies and the corresponding 3D defect information through scan data, which are also useful for downstream defect reconstruction and geometry-assurance tasks [19].

Unlike MVTec AD, MVTec 3D-AD and Real3D-AD, which are used primarily to benchmark and rank anomaly detection on a fixed set of object categories, the thesis does not use 3D-ADAM for anomaly detection benchmarking but rather the dataset’s defect masks paired with 3D scans as ground truth to build and validate 2D-to-3D mapping and a signed-deviation regression pipeline (Section 3.3), a use which is distinct from its original anomaly detection benchmarking purpose.

3D-ADAM’s pairing of RGB images with 3D scans and ground-truth defect masks makes it possible to evaluate whether defects can be identified and localized in 2D before being mapped onto geometry, which motivates the anomaly detection methods reviewed next.

2.3 Anomaly Detection

Identifying and localizing defects in 2D images is a hurdle when reconstructing those defects on the 3D part geometry. Before a defect can be reconstructed on the nominal CAD, its boundary and its type must first be established in the inspection image; this step is referred to as anomaly detection.

Anomaly detection is the task of identifying instances in a dataset that deviate significantly from a dataset’s expected or normal behavior. In the industrial context, anomaly detection specifically refers to detecting specific regions of an image or 3D scan that do not match the intended appearance of the defect-free product [11].

Additive manufacturing processes such as fused deposition modelling (FDM) are sensitive to defects like bulging, holes, surface warping, and over/under extrusion [5]. Since it is easy to acquire images of normal parts during production but costly and complicated to acquire images of every possible defect type, most state-of-the-art anomaly detection methods are unsupervised. They only train on normal samples and flag anything that significantly deviates from the normal at test time [23].

Representative 2D approaches include:

- PaDiM, which models the distribution of normal features at each spatial patch position as a multivariate Gaussian using a pretrained convolutional neural network [6].
- PatchCore, which builds a coreset memory bank of mid-level patch features extracted with a pretrained backbone and scores anomalies by nearest-neighbour distance, reporting up to 99.6% image-level AUROC on MVTec AD [23].
- CFA, which instead adapts a pretrained backbone’s features to the target dataset by learning a patch descriptor that pulls normal features into a compact hypersphere, reporting 99.5% AUROC on the same benchmark [15].

These 2D approaches were also extended to 3D and multimodal data. 3D PatchCore applies the same memory-bank and nearest-neighbour principle directly to point cloud features rather than 2D patches, even though the irregular, unstructured nature of point clouds makes consistent patch-level feature extraction harder than in 2D [16, 23]. M3DM, on the other hand, fuses 2D RGB texture with 3D depth and geometry information through a hybrid scheme to combine patch-wise contrastive feature fusion with a decision-layer fusion of multiple memory banks, reporting improved detection and segmentation precision over unimodal baselines on MVTec-3D AD [28].

The five methods represent the current state for unsupervised anomaly detection, but each produces either a binary/patch-level anomaly classification or an unlabeled anomaly-score map evaluated against single-object datasets captured under controlled conditions [2, 3]. The thesis requires converting a localized defect region into a signed, millimetre-scale deviation expressed on a specific nominal CAD geometry, which none of these anomaly detection outputs directly provide.

Since it has been established that the defects can be localized in 2D, the next obstacle is to separate the part from its background clutter in the image, which is addressed in the part segmentation below.

2.4 Part Segmentation

Part segmentation is a critical step during anomaly detection, as the background regions in the image inflate the anomaly scores and affect the precision. This thesis work considers the Segment Anything Model (SAM), which is an image segmentation model developed and released by Meta AI [12]. SAM overcomes some limitations of task-specific segmentation models with the motivation to create a universal segmentation model. The core idea of SAM is to generate valid segmentation masks from a variety of prompts and to generalise zero-shot to new image distributions and segmentation tasks. The SAM team built the SA-1B dataset, which contains over 1 billion segmentation masks on 11 million licensed and privacy-respecting images [12].

The SAM model is inspired by Natural Language Processing (NLP) and it is based on the idea of returning a valid segmentation mask when provided a prompt consisting of any information indicating what to segment in an image. SAM consists of three components that work together to execute the 'Promptable Segmentation Task'. The first component is an Image Encoder, which is a pre-trained vision transformer (ViT), which runs once per image to process the entire input image and produce a compact representation of the image in a form of rich image embedding. The thesis uses the ViT-B (Vision Transformer, "Base" size) variant of the encoder which is a mid-sized transformer that splits the image into fixed-size patches and processes them as a sequence, offering a practical balance between segmentation quality and inference speed compared to the larger ViT-L and ViT-H variants[12]. The second component is the prompt encoder, which encodes the user's prompt in the same embedding space. SAM considers two sets of prompt types: sparse prompts (points, boxes, text) and dense prompts (rough masks). The third component in the SAM model is the mask decoder, which is a transformer that efficiently maps the image embedding, prompt embedding, and an output token to a mask; a dynamic linear classifier that computes the mask foreground probability at each image location [12].

SAM is used here in its pretrained, zero-shot, automatic mask-generation mode without any fine-tuning on 3D-ADAM. This suits the present task, which only requires separating a single foreground part from cluttered industrial backgrounds across many previously unseen part geometries, rather than recognizing or classifying specific object categories.

With the defect localized and isolated from the background in 2D, the next step is projecting that 2D information onto the part's 3D surface, which requires registration of the scan to the nominal CAD, as discussed in the next section.

2.5 Registration

Registration is the process of finding the rigid transformation through rotation and translation such that two point sets expressed in different coordinate frames align into a common reference frame [4]. Registration is required because each structured-light scan is captured in the sensor's own coordinate system and covers only a

single partial view of the part, whereas defects must be expressed in relation to the nominal CAD geometry in order to compute a meaningful signed deviation. Without accurate registration, a correctly mapped 2D defect would be projected onto the wrong location on the 3D model, undermining both deviation measurement and geometry reconstruction.

2.5.1 Point Cloud Registration via Iterative Closest Point Algorithm

The Iterative Closest Point (ICP) algorithm is widely used for geometric alignment of two 3D point sets. Given a source and a target point cloud, ICP iterates three steps until convergence occurs. Firstly, for every point on the source geometry, it identifies the nearest point on the target geometry. Secondly, it computes the optimal transformation with rotation and translation with the aim of minimizing the mean squared error between the matched pairs. Lastly, it applies the transformation and repeats the procedure until suitable convergence occurs [4].

2.5.1.1 PCA-based Initial Alignment and Scaling

Principal Component Analysis (PCA) has been widely employed as an initialization strategy for point cloud registration. By computing the principal axes of each point cloud and aligning them, PCA provides a coarse initial transformation that places the source and target in approximately the same orientation and scale. This coarse alignment then serves as the starting point for ICP refinement. Recent work combines PCA-based initial alignment with a plane constraint to mitigate instabilities when a dominant planar surface is present, followed by multi-stage optimization for refinement [14].

In the context of dimensional inspection, PCA-based alignment is particularly useful when the source and target point clouds originate from different coordinate systems or measurement devices and thus differ not only in pose but also in scale. Per-axis scaling based on the ratio of bounding-box extents along each principal direction provides a straightforward mechanism for normalizing the scale before ICP is applied.

2.5.1.2 Reference Plane Alignment

In dimensional metrology and manufacturing inspection, parts are conventionally aligned to datum planes before measurement [24]. The primary datum is typically the base surface, i.e., the surface of the part that contacts the fixture or reference surface during measurement. Aligning the scanned point cloud horizontally provides a physically meaningful and repeatable initial orientation for subsequent registration steps [10].

The thesis pose estimation relies on the PCA- and symmetry-based strategies described in Section 2.5.1.1, tailored to each part's principal-axis geometry as detailed in Section 3.6.1. Reference-plane alignment illustrates a complementary, physically grounded initialization practice from dimensional metrology.

2.5.1.3 Registration Quality Metric

Registration quality must be verified. A standard research metric for comparing point sets is the Chamfer distance. For each point, the Chamfer-distance algorithm finds the nearest neighbor in the other set and sums the squared distances [7]. The chamfer distance metric is universally recognized and widely used for evaluation of point cloud registration, generative models, and shape completion as it is computationally effective and symmetric [29]. While chamfer distance is the standard research metric for registration quality, the thesis instead relies on a simpler tolerance-based measure, pose fit, defined as the fraction of scan points lying within a fixed distance of the CAD surface. It provides a fast, interpretable per-part registration quality indicator during pipeline validation, as detailed in Section 3.6.

However, the review of registration-related literature described above assumes either full-coverage scans or fixture-defined datum surfaces, which provide relatively well-constrained starting poses for ICP-based refinement [4, 10, 14]. The thesis uses each single-face scan of the part against a full solid CAD model of the nominal geometry. This leads to a highly under-constrained registration problem compared to the cases considered above. As a result, none of the discussed methods is applicable by itself to the described problem. To overcome this limitation, the pose estimation step described in Section 3.6.1 combines the coarse PCA- and symmetry-based initialization with an extra step of correction for front/back mirroring specifically designed for single-view partial scans.

2.6 PointNet and PointNet++

PointNet is a deep-learning architecture designed to reason directly about 3D point-cloud data. This neural network was designed to overcome the input-format limitations of conventional convolutional architectures. Typical architectures require a highly regular input data format and so the data is transformed into regular 3D voxel grids or collections of views before being fed into a deep-network architecture. This transformation renders the data unnecessarily voluminous and also introduces quantization artifacts that can obscure natural invariances of the data. PointNet is a unified architecture which directly takes point clouds as input and outputs either class labels for the entire input or per-point segment/part labels for each point of the input. The basic architecture of PointNet is simple yet effective as it uses a single symmetric function, max pooling. The network learns a set of optimization functions which select informative points of the point cloud and encode the reason for their selection. The final fully connected layers of the network aggregate these learned values into the global descriptor for the entire shape (shape classification) or are used to predict per-point labels (shape segmentation).

The classification network in PointNet takes 'n' points as input where each point has three coordinates. In order to make a model invariant to the input permutation, three strategies exist which are 1) sort input into a canonical order; 2) treat the input as a sequence to train an RNN, but augment the training data by all kinds of permutations; 3) use a simple symmetric function to aggregate the information from each point. The input transform stage first involves a mini-network called T-Net which

predicts a 3x3 transformation matrix which is applied to all input points in order to align the entire input set to canonical space before feature extraction. Each point is further processed independently through a shared neural network called multi-layer perceptron (MLP) which transforms the point into a 64-dimensional feature vector which encodes local geometric information. A second T-Net, included for feature transformation, predicts a 64x64 transformation matrix which is applied to all the 64-dimensional features. A regularization term is added for stable optimization and better model performance. A second shared MLP transforms the 64-dimensional feature of each point into a 1024-dimensional feature vector. A symmetric max-pooling operation aggregates information from all points into a 1,024-dimensional global feature vector which also makes the network invariant to the ordering of input points. The segmentation network inside PointNet concatenates the global feature with each point’s local feature to form a 1088-dimensional per-point feature. Again this combined feature is processed through a final shared MLP that produces per-point predictions [22]. PointNet ++ uses a PointNet hierarchy which groups points into local neighborhoods, extracts features within each neighborhood, and progressively aggregates them across increasing spatial scales through set-abstraction layers, with feature-propagation layers interpolating features back to the original points for per-point prediction. This hierarchical structure captures both fine local detail and broader context, making PointNet++ suited to per-point regression tasks on 3D surfaces such as the deviation prediction in this work [21].

Originally PointNet and PointNet++ were developed for the purpose of shape classification and semantic part segmentation on synthetic object datasets, for example, ModelNet and ShapeNet [22, 21] for assigning individual class labels to points or shapes. In this thesis, the per-point feature propagation of PointNet++ is used for continuous regression. Here instead of assigning a class label, PointNet++ predicts signed deviation magnitude at each point of a registered single-view industrial scan. This is a considerably different learning objective and data domain for the PointNet++ architecture than the one it was originally validated on.

The per-point regression capability of PointNet++ is used in this thesis for predicting signed deviation directly on the nominal CAD surface, with the aim of closing the loop from 2D defect detection to 3D geometry reconstruction, as discussed research gap in Section 2.8.

2.7 Learning-Based Geometry Reconstruction

The learning-based work on 3D geometry falls into four groups, which are discussed below.

CAD Reconstruction, also called reverse engineering, turns a point cloud or an image of an object into a CAD model. ComplexGen [8] rebuilds a CAD boundary representation (B-Rep) from a point cloud by predicting its vertices, edges, and surfaces together with how they connect. Point2CAD [17] does the same by combining segmentation with surface fitting, and Img2CAD [30] does it from images. These methods recover the clean design of an object, so small defects are smoothed away. They also do not have any reference model to compare against. By contrast, this thesis assumes that the nominal CAD is known and aims to recover the defect.

Shape completion recovers a complete point cloud from a partial one, because real scans are often incomplete due to limited sensor resolution, a single viewpoint, and occlusion. PCN [32] and PoinTr [31] predict a full point cloud from a partial scan. However, they fill in a plausible shape rather than measure how a part differs from a known one, so they do not capture part-specific defects.

Monocular depth estimation predicts a depth value for each pixel of a single 2D image, which can be converted into a 3D pseudo point cloud [33]. Some inspection studies estimate 3D information from a 2D image, for example, by generating a pseudo-point cloud from a weld image with monocular depth estimation and classifying the defect with a PointNet++-based network [33]. These methods provide relative depth estimates for one component type and no link to a nominal design. This thesis instead expresses the defect on the nominal CAD and outputs the deviated point cloud.

Geometric deviation prediction estimates how far a printed part will differ from its intended CAD shape, so that the CAD can be compensated before printing [34, 13]. Neural networks have been trained to predict how a printed part will deviate from its CAD, using CNNs [34, 9], graph networks on point clouds [13] and PointNet-based networks [18]. These methods are most closely related to this thesis and also support the use of a point-based network for per-point deviation. The difference is that they predict whole-part, process-related error from CAD and process data, mainly to compensate the CAD before printing. They do not use 2D inspection images or localised defects such as bulges, holes and cuts.

Approach	Input	Output	Uses nominal CAD?	Defect-aware?
CAD reconstruction (ComplexGen, Point2CAD, Img2CAD)	Point cloud or image	Clean parametric CAD	No	No
Shape completion (PCN, PoinTr)	Partial point cloud	Completed shape	No	No
Depth from image (weld inspection)	2D image	Height map / pseudo point cloud	No	Partly
AM deviation prediction (Zhu, Huang, GraphCompNet)	CAD and process data	Whole-part deviation field	Yes	No
This thesis	Per-point signed deviation and nominal CAD	Deviated point cloud	Yes	Yes

Table 2.1: Current thesis compared with related learning-based approaches

In short, the reviewed methods either recover an idealized design, complete a shape, estimate relative depth, or predict process-related error. None of these methods takes 2D defect information and produces the defect deviation on the nominal CAD of an individual part. For this reason, they cannot be reused here, which defines the research gap below.

2.8 The Research Gap

Section 2.1 indicates that the current defect-inspection methods rely either on operator-based manual assessment or on 3D scanning, which is accurate but slow and whose scalability depends on part geometry [1, 5, 25, 26]. The anomaly-detection literature

reviewed in Section 2.3 shows that, without using a 3D scan, the defects can be localized in 2D images [11, 23] together with a part segmentation method to isolate the part from the background as mentioned in Section 2.4 for proper anomaly scoring [12]. The registration method mentioned in Section 2.5 along with the point-based deep-learning network from Section 2.6 enables expressing and predicting deviation on a nominal CAD once a 2D-to-3D mapping path exists [4, 22, 21]. Nevertheless, these methods from Section 2.1-2.6 solve various aspects of the problem, but their contribution to the whole problem seems disconnected. Learning-based reconstruction work recovers clean designs [8, 17], completes shapes [32, 31], estimates relative depth [33], or predicts process-related error [34, 13]; however, to the best of our knowledge, none of the existing methods predicts physical severity of defects directly from the output of 2D anomaly detection without using a 3D scan at inference time. Bridging this gap would enable rapid quality screening by estimated severity, reserving 3D scanning for borderline cases, and this is the exact gap addressed by this thesis.

3

Methodology

3.1 Research Methodology

This thesis follows the Design Research Methodology (DRM) framework shown in Figure 3.1, which consists of four stages, namely Research Clarification, Descriptive Study I, Prescriptive Study and Descriptive Study II. This framework is appropriate for the research that aims to improve product-development practice by first understanding the problem, studying the existing situation, developing a solution and finally evaluating it. These four stages in this thesis are applied as follows.

The product development problem was identified in the Research Clarification stage: current techniques for defect inspection and 3D geometry reconstruction in additive manufacturing can be improved, particularly in mapping 2D defect information to 3D geometry. This stage of the framework established the motivation and the goal of the thesis.

The Descriptive Study I stage included a literature study of areas relevant to this thesis, such as defect inspection, available datasets, anomaly detection, 3D registration, and geometric deep learning. This study, which is presented in Chapter 2, clarified the current state of the field and helped in positioning the proposed approach.

In the Prescriptive Study stage, the proposed solution was developed in the form of a pipeline for defect mapping, deviation prediction, geometry reconstruction and quantification. This stage represents the main methodological contribution and is explained further in this chapter.

The last stage, Descriptive Study II, evaluated the developed pipeline against the dataset, reported its performance and defined the limitations, which are presented in Chapter 4 and Chapter 5. The information required for scan-free defect prediction at inference was identified in this stage which will help in the refinement of this pipeline in future work.

3.2 Overall Pipeline Architecture

This section gives an overview of the developed pipeline in the Prescriptive Study stage mentioned in the DRM framework (Figure 3.1). The pipeline reconstructs defect-aware 3D geometry from a defect annotation and a nominal CAD model of the part and is organized in phases as shown in Figure 3.2. The 2D-to-3D mapping phase performs 3D lookup and extracts 3D points and overlay images. The first phase re-applies a correspondence verified in Phase 0 and registers each scan of the part to its

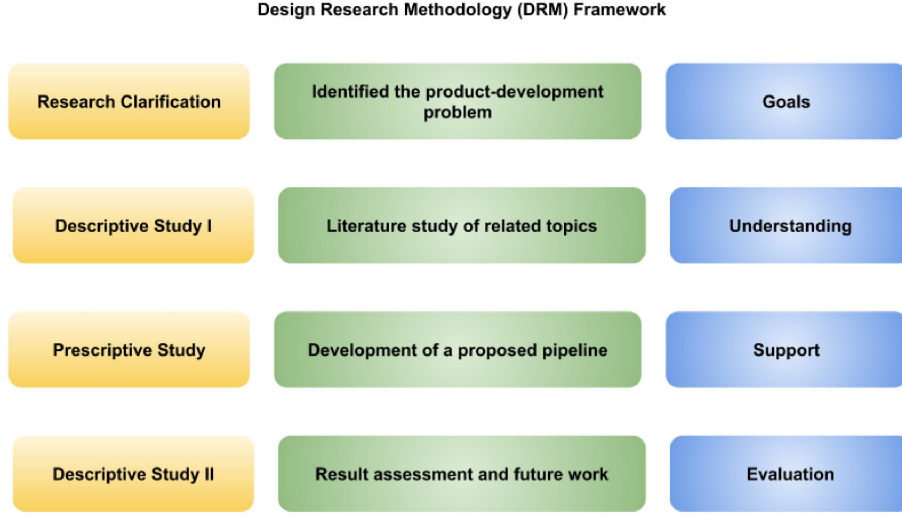


Figure 3.1: Design Research Methodology Framework

respective nominal CAD. It also involves computing the signed deviation between a defective scan and a nominal scan. This produces labeled data. The outcome of this phase becomes the training data for the subsequent model, comprising NPZ files and one coarse-heatmap NPY file per sample. In the second phase, the pipeline splits into two branches to evaluate two formulations. In the training branch, a PointNet++ model is trained to predict the signed deviation from provided input and outputs a set of learned weights. In the inference branch, the model predicts deviation from the test-split input, and the branch outputs prediction-performance metrics and comparisons between the two formulations and against the baseline. Both branches merge again in the third phase of the pipeline which involves defect-aware geometry reconstruction on the nominal CAD model using the measured and predicted deviations separately for evaluation. The last phase of this pipeline is quantification, in which defect severity is expressed using three geometric proxies and exported to a CSV file. This structure explicitly demonstrates that the scan is required to generate training data, whereas the scan-free inference branch is the part of this pipeline whose feasibility this thesis attempts to investigate.

3.3 Data Preparation

Prior to building the reconstruction pipeline, the 3D-ADAM dataset was explored in detail to understand the general composition, distribution of defects, and suitability for the proposed pipeline workflow [19]. In this stage, a Python script was used to examine the dataset part by part, to determine the number of available training, validation, and defective test samples, the number of defect categories occurring in each part, and which parts offered the greatest defect variety. Also, for each part, the counts of defect images and depth maps and the availability of synchronized pairs were recorded. Parts were further ranked by variety in defects and availability

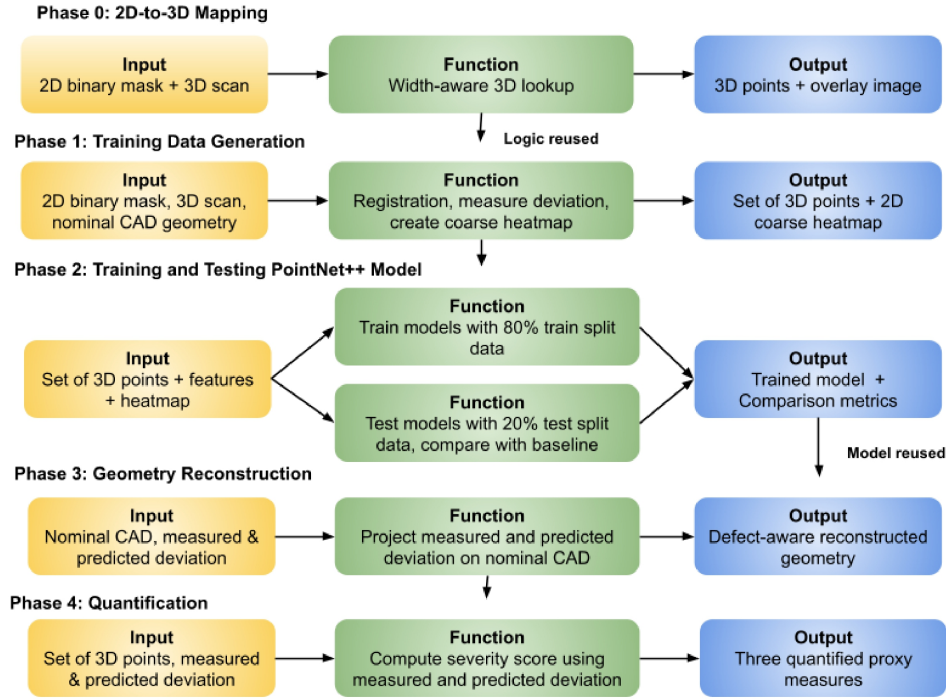


Figure 3.2: Proposed Pipeline Architecture in the form of function blocks for each phase

of samples for selection of a representative part for the initial pipeline development. This preliminary exploration was necessary because out of multiple industrial parts with defect variety offered by 3D-ADAM, selecting a part with diverse defects, synchronized RGB and 3D data, and high-quality annotations was essential as a stable starting point.

The exploration indicated the structured hierarchical organization of the dataset with individual folders for training, testing, validation and sensor-based data, including point clouds, RGB images and annotations. The dataset is organised by part instance and condition, so that each scan can be traced to its physical part, its nominal or defective state, and the sensor used to capture it. This thesis uses only the data from the MechMind-Nano sensor. The RGB image, point cloud and the annotation masks for each part are in one-to-one correspondence for a given view. Two annotation types from the dataset are used: a part-segmentation mask, which separates the part from the background, and a defect-segmentation mask, which provides pixel-level ground truth of the defect region. This 3D-ADAM dataset also provides bounding-box and machine-element annotations.

Based on the ranking of parts, part 3M2C was selected for the initial implementation. Part 3M2C offers five types of defects relevant to additive manufacturing, namely bulge, cut, hole, over-extrusion, and under-extrusion along with synchronized RGB and 3D depth data of 48 nominal images for training as shown in Figure 3.3, 16 validation images, and 81 defective images for testing. While another part was ranked higher than 3M2C due to its marginally higher overall score, 3M2C was

selected because it offered a balanced combination of defect variety in both mechanical and process-related anomalies, and sufficient samples for training and testing. The 3M2C was selected because it met the criteria for a stable and sufficiently diverse part.

The 3D-ADAM dataset provides the sensor-captured data for 23 parts. Of these, the nominal CAD geometry (STL) which serves as a fixed base in later stages was available for only 16 parts and hence only those parts were included in the pipeline. The initial work on 3M2C led the way for the pipeline’s extension across these 16 parts in the dataset whose nominal CAD geometry (STL) and complete data were available. Each part contributes between 32 and 96 nominal images for training and 16 to 96 defective images for testing, spanning six defect categories overall: bulge, crack, cut, hole, over-extrusion, and under-extrusion. For the full pipeline, a subset of these 16 parts was processed and the criterion for selecting this subset is explained further in Section 3.6.1, as it relates more closely to registration feasibility than data availability.

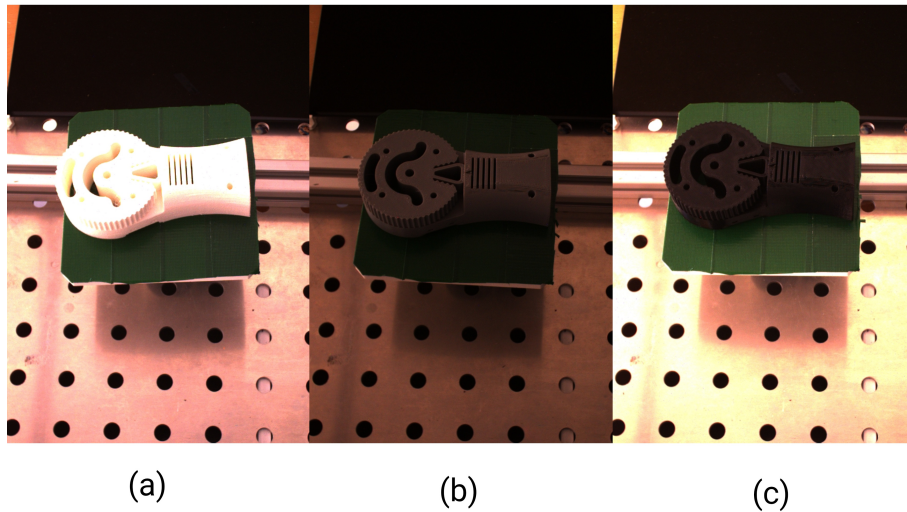


Figure 3.3: RGB images of all three instances of sample 006 of part 3M2C with nominal condition from 3D ADAM dataset [19]

3.4 Anomaly Detection and Part Segmentation

The automated anomaly detection and part segmentation stage was explored as a possible front-end for the pipeline. This section summarizes this exploration and explains that the dataset annotations were ultimately used instead of outcomes of this stage. Hence it is included here for the purpose of completeness rather than as an actual stage of the pipeline.

3.4.1 Anomaly Detection

Across three modalities, five methods of anomaly detection were tested with the Anomalib subset for part 3M2C. Three methods were based on 2D RGB images. The first method, PaDiM, uses a three-layer ResNet-18 as the backbone and models the statistical distribution of normal features at each spatial position in the image. PatchCore utilizes a WideResNet-50 backbone which extracts normal patch features, stores a coreset in a memory bank and scores test patches using nearest-neighbor distance. The last one, CFA (Coupled-hypersphere-based Feature Adaptation), trains a small adapter network which transforms general features from the backbone into a 256-dimensional space where nominal features cluster tightly around a learned centre, whereas defective features fall away from it. Another method was based on 3D geometry: a 3D variant of PatchCore operated on the sensor depth maps and extracted simple geometric descriptors (mean depth, standard deviation, surface gradients) from each patch due to the unavailability of a large pre-trained 3D network backbone. Finally, a 2D-3D fusion method was tested in which both 2D image and 3D depth modalities were combined using score-level fusion which included normalization and averaging of each modality’s independent anomaly scores.

Among these methods, the 2D-image-based PatchCore achieved the most relevant image-level AUROC score on 3M2C. Nevertheless, the performance of the methods in terms of anomaly detection was not trustworthy enough to be utilized in the pipeline as the results were highly dependent on critical hyper-parameters such as the size of PatchCore’s coreset and fusion weighting. As improving anomaly detection was outside the scope of the thesis, these methods were not taken further into the pipeline.

3.4.2 Part Segmentation using SAM

During manual observation of RGB images from the 3D-ADAM dataset it was noticed that the images contain background elements from the setup such as pegboard surface, metallic rails, and the platform on which each part rests. All these background elements are visible alongside the part in every image. During anomaly detection, it became evident that these background elements were actually confused with normal appearance features and affected the anomaly score, degrading the anomaly detection on parts with a large portion of the image occupied by background elements. To resolve this, the Segment Anything Model (SAM) was implemented to obtain part masks for each image. SAM was utilized in automatic mask-generation mode; the mask area covering the central region of the image was considered the largest predicted mask and selected as the part mask based on the observation that almost every part lies in the central region of the image. This SAM segmentation was applied to all parts, and the resulting binary masks separated the part from the background across all training and test images.

3.4.3 Use of Dataset Annotations

The above-mentioned anomaly detection and part segmentation methods were tested, but they were not included further in the pipeline due to insufficient reliability and

their improvement being outside the scope of this thesis. Instead, the ground-truth defect masks and binary part masks provided in the 3D-ADAM dataset and shown in Figure 3.4 were used in the pipeline, because they were provided by experts and also provide a reliable basis for the 2D-to-3D mapping and reconstruction work central to this thesis. This decision was made with the supervisor’s approval, to focus the effort on the actual work included in the scope of this thesis: mapping, deviation measurement, and defect-aware geometry reconstruction.

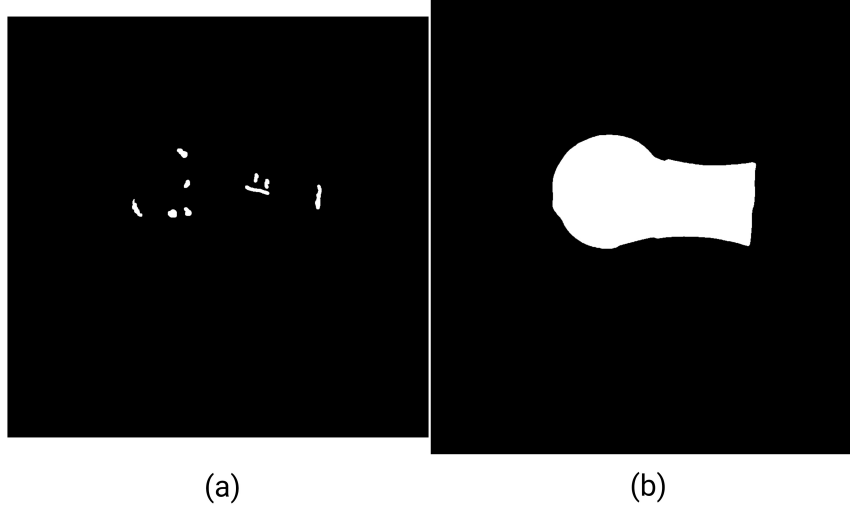


Figure 3.4: Sample 006 of part 3M2C (a) Ground-Truth annotation of Bulge defect (b) Binary Part Mask

3.5 Defect-to-Surface Mapping

The purpose of this mapping was to convert 2D defect annotations from 2D images into 3D points on the 3D part surface according to the process shown in Figure 3.5.

3.5.1 Mapping

The MechMind-Nano sensor produces an organized point cloud, which is a 2D grid of $H = 1024$ rows and $W = 1280$ columns where each grid element (r, c) already stores a 3D coordinate $P_{r,c} = (x, y, z)$. Due to this dataset structure, mapping a 2D mask pixel to its 3D point is a direct array lookup rather than a separate reconstruction. For a defect mask pixel at column u , the corresponding 3D point can be expressed as

$$p = G(r, u + \delta) \quad (3.1)$$

where G denotes the point cloud grid and δ is a column offset correcting a width mismatch between the defect mask and the grid. The defect masks are annotated

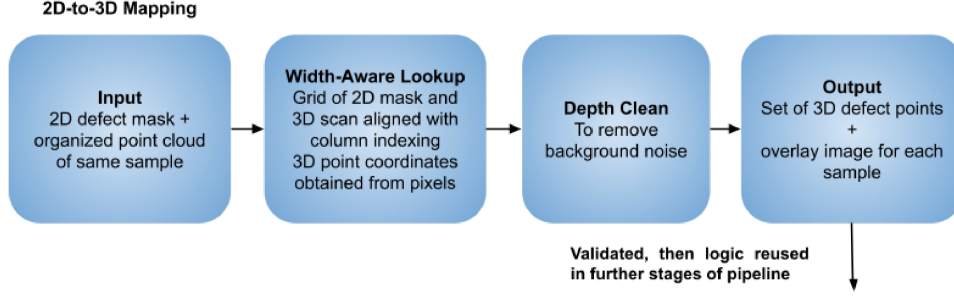


Figure 3.5: Architecture of 2D-to-3D Mapping stage

on a centered crop of width $W_m = 1024$, while the grid has width $W = 1280$. Hence,

$$\delta = \frac{W - W_m}{2} = \frac{1280 - 1024}{2} = 128 \quad (3.2)$$

The part masks with full width match the grid width without requiring any offset. ($\delta = 0$). After lookup, the obtained 3D points are depth-filtered to remove background bleed which appears through holes and dropout regions. This cleaning retains only those points within a band around the median depth z_{med} :

$$|z - z_{\text{med}}| < \tau_z, \tau_z = 0.045 \text{ m} \quad (3.3)$$

The operations involved in this step are parsing the organized point cloud into the $H \times W \times 3$ grid G ; the width-aware pixel to 3D lookup of Equation 3.1; and the depth cleaning filter of Equation 3.3.

The output of this step is a set of 3D defect points and an overlay image per sample which is formed by redrawing the mapped points on the RGB image of the corresponding sample for visual confirmation. These outputs were created to validate the mapping method but they are not themselves fed into later stages but rather the same lookup logic (Equation 3.1 and Equation 3.3) is re-applied within the deviation measurement stage to extract 3D points from binary part masks. Hence this mapping stage serves as method validation and not as a source of pipeline data.

3.5.2 Verification

In addition to the visual check given by the overlay images, the mapping was verified quantitatively using measures that avoid reusing the mask being checked. Three measures were calculated; two of them coincide, as explained below. The first measure, valid fraction, calculates the fraction of the defect mask pixels which actually land on a real 3D surface point. A valid point is finite, nonzero, and within the depth-band criterion. Valid fraction is calculated by dividing the number of valid 3D points by the number of mask pixels. This measure drops the pixels that map to background or dropout regions that were not captured by the sensor and shows a low value.

The second measure, round-trip IoU, takes the valid 3D points, projects them back onto the 2D grid, and calculates the intersection-over-union between these projected back pixels and the original mask. Because the 2D-to-3D mapping is an exact array lookup, every valid pixel projects back to its original location. Hence this measure coincides with the valid-fraction measure because both reduce to the same quantity. This is a meaningful agreement because a wrong crop or column offset would take the reprojected points off the mask and reduce both values at the same time. Hence the high values of these two measures assure the absence of such coordinate error. The third measure, mapping on-part fraction, is an independent measure which tells the fraction of defect 3D points within 1.5 mm of the part surface, where the part surface is built from the binary part mask annotation not used in producing the defect mapping. This fraction is computed by nearest-neighbor distance from each defect point to the point cloud of the part. This is a critical measure because it is computed using independent information, so it catches errors such as a wrong crop or offset shifting the defect off the part; such an error appears as a drop in the computed value. Hence this mapping on-part fraction is primary evidence of correct defect mapping.

3.6 Pose Estimation and Signed Deviation Computation

The purpose of this stage in the pipeline was to generate the training data for the deep geometric-learning model. This stage involves registration of the sensor scan (good PLY) to the nominal CAD of the part followed by measuring the signed deviation caused by the defect. The inputs to this stage include the defect scan (defect PLY) and corresponding good scan (good PLY) of each sample, the nominal CAD geometry (STL file) of that part and the mapped defect region of each sample. The scans also contain background elements along with the actual part, so the first step is part isolation from the scan using binary masks. The binary masks from the dataset select the part points, which are then cleaned with a depth band filter which keeps the nearby points to the part’s median depth and a depth-band filter, which removes background bleed, and a k-nearest-neighbor trim, which removes isolated outliers. The part’s nominal STL, shown in Figure 3.6(a), is a triangle mesh and for registering a point cloud against it, the STL is sampled into a dense point cloud using an area-weighted surface-sampling method which involved sampling around 50,000 points distributed across the triangle faces where each triangle t received points in proportion to its area A_t and relative to the total surface area for even distribution and maintaining uniform point density regardless of triangle size as shown in Figure 3.6(b). The PLY scan is in meters while the CAD geometry is in millimeters; hence the scan is scaled by 1000 to match the CAD. This sampling was performed once at the start and it then serves as the registration target and the fixed substrate (nominal geometry) stored in every training sample of that part.

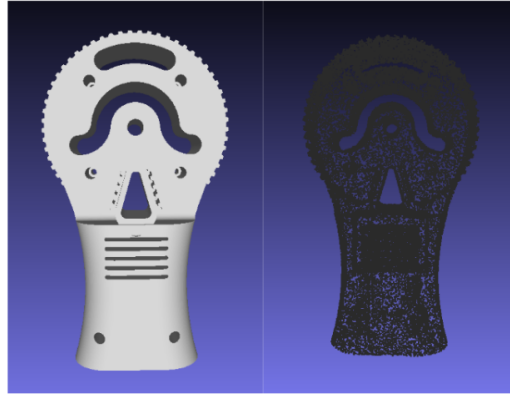


Figure 3.6: Part 3M2C (a) Nominal STL Geometry (b) Point Sampled Geometry

3.6.1 Pose Estimation

Every scan in the dataset captures only a single partial face of the part while the nominal CAD describes the entire solid part. Registering that single face to an entire solid is under-constrained; hence a coarse-to-fine approach is used where rough initial candidate poses are generated and then each pose candidate is refined and the worst-fitting candidate is eliminated iteratively to finally select the most suitable pose. The initial coarse orientation depends on the part’s symmetry so it genuinely changes between parts. Hence three initialization strategies were used as shown in Figure 3.7, selected for each part according to its bounding box and symmetry.

The first strategy is used for parts with three different principal-axis lengths such as 3M2C, 2M2H, 2M2M. The initial orientation is obtained from the part’s principal axes. A near-planar face of the part fits the solid equally well in two opposite orientations which introduces a front/back ambiguity which further results in mirror ambiguity for defect placement. Hence a front/back correction is applied where the candidate pose whose raised relief features match best is selected because such features exist on only one face.

The second strategy is used for round, disc or nearly cubic parts having similar principal-axis lengths such as 1M2, 1M3, 4M2. Here the principal axes are unable to fix the in-plane rotations; hence the dominant surface normal is used as an anchor, and all candidate rotation angles are tested and the one with best-fitting is selected.

The third strategy is used for a nearly symmetric part such as 3M1, which is prone to locking into a flipped pose. So the random starting orientations are combined with an angle sweep to reliably find the correct pose.

In all three strategies, each coarse pose candidate is refined by the trimmed Iterative Closest Point (ICP) method mentioned in Section 2.5, which eliminates the worst-fitting point pairs at every iteration to mitigate the contamination of the alignment due to single-sided scans.

The front/back mirror correction method mitigates the mirror ambiguity mentioned above. For a near-planar face which fits the solid equally well on either of its faces, the possibility of the defect getting placed on the wrong side increases. Hence, the

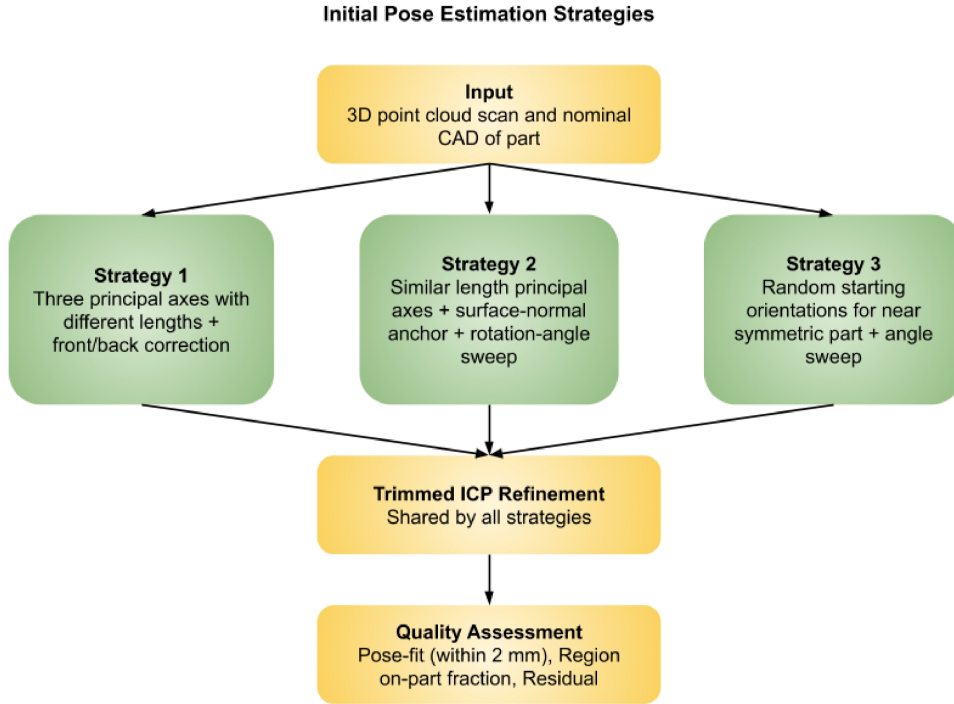


Figure 3.7: Three Initial Pose Estimation Strategies used for Registering 3D Scan to CAD

correction method uses raised relief features which exist on only one face, so each candidate pose is weighted based on the distance between each point and the part’s mid-plane and the pose whose relief feature matches most tightly is selected. This relief-weighted scoring lowered the mirror-flip rate from 2 of 8 to 0 of 8 in controlled single-face tests. On noisy partial scans, the mirror ambiguity is reduced but not completely eliminated, and remains as a characterized limitation.

The dataset does not provide any ground-truth pose for verification so the registration quality is assessed by proxy measures. Pose-fit indicates the proportion of scanned points within a tolerance $T_{\text{fit}} = 2 \text{ mm}$ of the CAD after alignment. The 2 mm tolerance considers scan noise and the partial view registration while differentiating the aligned points from misaligned points. Residual is the median distance from scan points to the CAD surface. A third proxy called region on-part fraction is defined below in Section 3.6.4.

Registration was done on the parts where a reliable automated pose could be established. Out of the 16 parts from the dataset with CAD geometry, seven parts were processed for the full pipeline. The part’s geometry was a determining factor in this decision as the automated pose estimation depends on differentiating between the part’s faces and resolving its orientation, which requires sufficient asymmetry and distinctive relief of the part. Disc-like parts are highly symmetric, and their geometry does not provide the required features for reliable orientation and registration. Manual intervention or specific handling for parts would be required to correct such cases which was not possible within the limits of this thesis. The seven parts that

were processed were selected as their geometries reliably registered with the three pose estimation strategies mentioned above, covering all six defect types. Before this automated approach, the manual selection of correspondences between a 3D scan from the dataset and the nominal CAD was attempted as shown in Figure 3.8 but it proved unreliable and non-scalable across the dataset which motivated the above-mentioned automated approach.

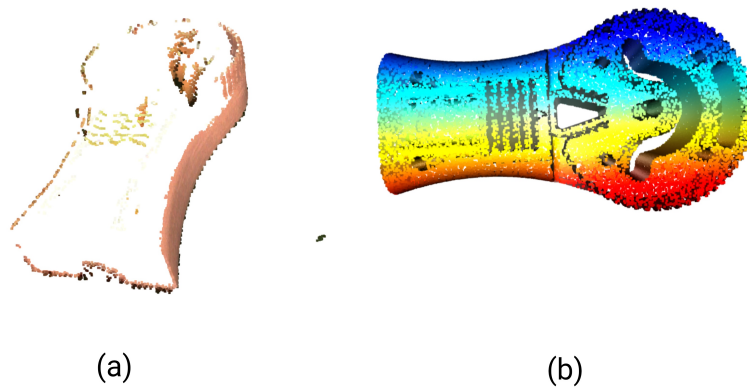


Figure 3.8: Manual Correspondence attempt for pose estimation by choosing 3D points on (a) PLY Scan from dataset (b) nominal CAD

3.6.2 Signed Deviation

The deviation magnitude is measured between the defective and corresponding defect-free scans of the same sample and in the same scanner coordinate system. The good scan is registered to the defect scan using the Iterative Closest Point (ICP) method described in Section 2.5. Because both scans are in the same coordinate system, the magnitude of deviation is pose-independent, meaning that it does not depend on the registration of the 3D scan to the CAD. The deviation magnitude is the per-point distance between the two scans over the defect region. The defect sign is assigned by defect physics, positive for material-adding defects such as bulge and over-extrusion and negative for material-removing defects such as crack, cut, hole, and under-extrusion.

The two physical scans of the same part are never identical, because of the slight deformation in the overall shape of that part between the two printed instances. This deformation is a smooth, dimensional difference between the two prints across the whole width of the part. To avoid misinterpreting this difference as a defect, a local baseline is subtracted at each defect point. For a defect point p with raw measured deviation $d(p)$, the baseline $b(p)$ is the median deviation of its surrounding

k nearest nominal surface points, and the corrected deviation is

$$d'(p) = d(p) - b(p) \quad (3.4)$$

This subtraction acts as a high-pass filter, removes the low-frequency (smooth, part-wide) difference between two prints and retains the high-frequency (sharp, localized) defect signal. The neighborhood size ($k = 50$) was selected to be large enough to ensure reliable determination of the local surface while small enough to keep the baseline localized and not covering the part curvature.

The effect of the high-pass filter was quantified during iteration on part 2M2M. Before this correction, the original good versus defect comparison classified 36% of the total part surface as deviating by more than 2 mm, which was physically implausible for a localized defect. Inspection determined that the reason was a smooth global difference between two prints rather than any defect. After this correction was applied, the affected fraction decreased to 4% for that part, without affecting the localized defect. This high-pass filter does not affect parts with negligible difference between two prints, so it was then adopted as the default for all parts.

The signed deviation is then projected onto the nominal CAD using the pose. The nominal points within a region radius $r_{\text{reg}} = 3$ mm of any projected defect point are labeled as *region* = 1 and assigned the deviation of their nearest defect point, while all other nominal points are labeled *region* = 0. This produces a per-point deviation field that is used as the training target in a later stage of the pipeline.

3.6.3 Coarse Heatmap Generation

The computed signed deviation encodes both the direction and the magnitude of each defect. The main focus of this thesis is to verify the possibility of providing this information to the learning model. That form of information should allow direction to be inferred, which nominal geometry alone cannot support; it should also resemble information producible by an image-based detector. For this purpose, a coarse 2D heatmap, illustrated in Figure 3.9, was derived for each sample from its measured deviation. To construct the heatmap, the points representing the defect region are projected onto a 2D grid with the principal axes defining the two longest extents of the part. The grid resolution was intentionally fixed at 64×64 to make it coarser than that of the deviation field at each point. The deviation magnitude for each cell of the grid is quantized into discrete severity levels by defining the maximum severity for the points that fall into the cell with the help of thresholding limits. The severity map is then smoothed using a small averaging kernel, giving it the appearance of a diffuse anomaly field instead of a set of distinct points. The second channel encodes the defect sign based on defect category (+1 for added material, -1 for removed material, 0 for nominal). The outcome of this step is a two-channel array including coarse defect severity value and a sign value.

The coarse heatmap is an intentional design decision, because if the model that maps the heatmap back to the deviation field is provided with the heatmap encoded with exact per-point deviation, then the model will not learn or demonstrate anything but will just perform simple recreation of its own input. Moreover, it would be impossible to create an exact-deviation input without a scan, which would make



Figure 3.9: Coarse 2D Heatmap of part 3M2C (a) Sample 001 with Under-Extrusion (material removing) defect (b) Sample 014 with Over-Extrusion (material adding) defect

the purpose of scan-free inference pointless. The coarsening of the heatmap into a binned, smoothed, low-resolution severity map defined by the sign derived from defect class resembles an output that could plausibly be generated by a 2D anomaly detector. The model’s task of translating a rough 2D severity map into 3D deviation field on the CAD also becomes meaningful. This representation also demonstrates the connection to scan-free operation as in this work the heatmap is derived from the measured deviation which is computed from scan but in an actual deployed system it would be generated through RGB-based anomaly detection, thereby eliminating the need for a scan during inference.

This design choice results in two consequences. First, because the sign is encoded in the heatmap supplied to the model, the model is expected to predict direction of the deviation with a high degree of accuracy; this is a consequence of the sign encoding in the input rather than evidence that the model predicts direction from geometry, and is reported as such. Second, the thresholds set for severity binning are correlated with the deviation scale. To ensure consistent severity meaning between parts, the thresholds were calibrated according to the deviation range after the high-pass filter was applied across all parts.

3.6.4 Quality Filtering

The output of pose estimation was an NPZ file produced for every processed sample containing the nominal points, the region labels and the deviation values, along with a per-part manifest recording each sample’s quality measures and a good/not-good flag.

Two indicators were used as the primary quality gates: pose-fit and the fraction of region on-part, with a median residual as a backup criterion to discard poses that passed the overlap check but registered loosely. The region on-part fraction is the proportion of a sample’s defect points that fall within the region radius r_{reg} of the

CAD after registration, a test that the defect was placed on the part. This on-part fraction metric is distinct from the mapping on-part fraction mentioned in Section 3.5.2.

The thresholds were varied for each part rather than as fixed constants for two reasons. First, the possible pose-fit varies based on part geometry for example, a flat disc viewed at an angle cannot achieve the overlap which is possible to achieve by a well-featured part, so the threshold of pose-fit for such a disc should be lower. Second, the on-part threshold was determined by the intention to match defect depth, because ideally, a deep bulge or hole lies far off the nominal surface, so a deep-defect part like 1M3 required a lower on-part threshold than a shallow defect part. The baseline part 3M2C, which registered reliably, used an on-part threshold of 0.60 and pose-fit threshold of 0.35 while more difficult parts used some relaxed threshold values such as on-part in the range 0.15 – 0.35 and pose-fit in the range 0.25 – 0.50.

The parts which showed fewer good samples despite relaxed thresholds were reported as such and not compensated for by relaxing the thresholds further, so the sample quality was not traded for increasing sample count. Seven of the 16 parts were selected for the complete pipeline because their pose-estimation strategies produced reliable registrations, as mentioned in Section 3.6.1.

3.7 Deviation Prediction

This stage investigated whether a deep geometric learning model could predict the deviation field on the nominal CAD geometry based on the defect region, with the longer-term motive of eliminating the need for a 3D scan at inference.

PointNet++, a hierarchical network of set-abstraction and feature-propagation layers built in PyTorch without custom CUDA operators was chosen here to ensure portability on the HPC cluster. As shown in Figure 3.10, in the first stage of set-abstraction, the points are progressively subsampled, and the features of neighboring retained points are combined through a shared multi-layer perceptron (MLP); successive layers thereby capture geometric structure at increasing spatial scales. In this particular thesis work, the first layer reduced the set of points to 512 points with 128-dimensional features, and the next layer to 128 points with 256-dimensional features. The second stage of feature propagation interpolated these features back to the original resolution, combining the current layer’s features with the skip features from the corresponding abstraction layers thereby retaining both fine local detail and broader context. Finally, a per-point head mapped every point’s features to a single signed-deviation measure. This network took as input the coordinates of each point along with its region flag and for the heatmap-enabled model it also took two more channels which were the coarse severity and the sign of the defect.

The formulation was reached through many iterations. In the beginning, a model was trained on a single part 3M2C to predict signed deviation but the prediction was not useful due to a constant-sign predictor as defects of only one part are sign-imbalanced. Based on the assumption that including both material added and removed defects would allow the model to learn direction, pooling all parts was tested but still sign accuracy remained near chance. It was discovered that the di-

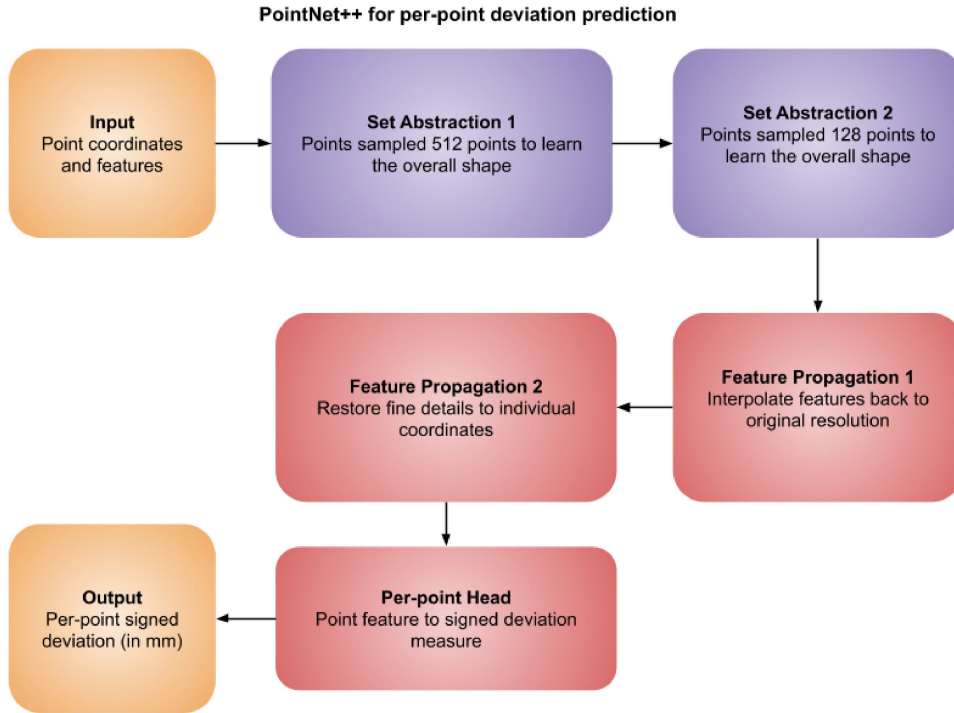


Figure 3.10: Architecture of PointNet++ network

rection of deviation cannot be determined based on the input of the model, because the information required for sign prediction is not available in the nominal CAD geometry where the surface remains identical regardless of whether material is later added or removed at a given location. Another magnitude-only model, where defect sign is assigned externally based on defect class, was considered a possible solution but not adopted as the final approach, because this model did not recover the sign from data and so did not remove the limitation.

These findings led to the formulation of a heatmap-augmented model. The coarse heatmap is provided as an additional per-point input to avoid external sign assignment. Each nominal point received the coarse severity and sign values from the heatmap based on its projected grid location along with coordinates and region flag. The model hence directly predicts the signed deviation based on the provided input consisting of a 3D point, the binary region flag, the coarse severity and the sign. The first feature propagation layers and input dimensions of the network were adjusted to accept these additional input channels.

The two formulations, the pooled geometry-only model and heatmap-enabled model, both predicting signed deviation were compared on identical data to understand the effect of the heatmap input. Both models were trained on the 348 good samples pooled across the seven parts, split using stratification by defect type with an 80/20 for training/test split; training ran for 80 epochs. The model was trained with a batch size of 8 and 4096 sampled points for each cloud, with a region-weighted mean-squared error loss weighting defect-region points approximately ten times the nominal majority points. Both models achieved their maximum performance near

20 epochs, so epoch 80 was selected to provide a consistent margin after convergence and results from that epoch are reported.

The evaluation was conducted based on magnitude MAE value against a class-average baseline and sign accuracy for each defect type. The high-pass filter leaves many region points near zero where sign is not physically meaningful and becomes noise; hence sign accuracy was calculated over significant points with deviation magnitude above 0.30 mm with the coverage fraction also reported. In the Results chapter, more information regarding this comparison, magnitude and sign is provided. Briefly, after providing the heatmap input, magnitude error decreased relative to the geometry-only model and to the class-average baseline in all six defect types and the sign of the deviation was also recovered on significant points. Further, the Discussion chapter includes interpretation of the results and also explains the reason for the high sign accuracy which is because the model rapidly converges while learning a relatively shallow refinement of a sign-informative input.

3.8 Geometry Reconstruction

This stage transforms the measured or predicted deviation field into reconstructed 3D defect-aware geometry, where the part surface is displaced as per the respective defect. The nominal CAD geometry of the corresponding part is used as a base. For each sample, the batch process reads the measured deviation from the NPZ file, computes the predicted deviation using the trained model, and applies each field separately to the nominal CAD model of the part. This operation involves nearest-neighbor interpolation, mesh subdivision and file writing which are geometric in nature and so it is performed on the CPU without requiring GPU acceleration.

In this process, for every sample, a surface normal is computed at each nominal point of the part by analyzing local principal components of its k nearest neighbors. The normal is considered the eigenvector associated with the smallest eigenvalue of the local neighborhood covariance, and it is oriented outward from the centroid of the part. After that, every point on the part surface is displaced along its normal based on the signed deviation; thus the points protrude above the surface for material-added defects and for material removed defects the defect points recede into the surface. A displaced point cloud with the defect in three-dimensional shape was produced from this process. By default, a true scale was used for applying displacement in which one millimeter of deviation corresponds to one millimeter of displacement while an additional exaggeration factor may be used for the purpose of clear visualization.

This updated approach of displaced point cloud was pursued after the earlier reconstruction method was found to be inadequate. Initially, the defect was projected by displacing the nominal triangle mesh vertices along their normals and exporting the deformed mesh. The coarse and non-uniform triangles of the nominal mesh could not smoothly show the local deformation hence the visible triangulation artifacts were produced in this approach. Another approach of a color-only point cloud where the defect points were marked by color without displacement was also considered but not implemented further because the defect was not represented as physical geometry. The final approach in which each point is an independent point

cloud was displaced along its normal and a separate surface-reconstruction step was subsequently adopted because it avoided triangulation artifacts and represented the defect as actual surface geometry.

Two forms of output are produced for each deviation field in this stage of the thesis. The primary output is the point cloud displaced as per deviation which is exported in PLY form with per-point normals and signed deviation coloring. This is considered the primary form because it does not include any mesh connectivity which further might show any triangulation artifacts after direct deformation of nominal mesh. The availability of per-point normals enables surface reconstruction of this point cloud into a watertight mesh with the Screened Poisson method in a standard mesh-processing tool which can further export it in STL format for CAD inspection. The other form of output is a deviation map where the nominal surface is colored according to deviation as a scalar field and exported in VTK format for visualization. For quantitative color-mapped inspection, this form of output is suitable because no surface-distortion artifacts are possible here as direct deformation is not applied. However, because deviation is interpolated across the faces of the nonuniform nominal triangle mesh, the color-mapped region within large triangles may appear faceted. This only affects the visual appearance of the color map and not the deviation or the actual geometry, and the point cloud remains the primary output.

During surface reconstruction with Screened Poisson, a limitation was observed. This method fits a smooth watertight surface throughout the whole point cloud, hence in some areas of the part with few sampled points or noisy normals such as at boundaries or around thin nominal features, this method reconstructs a surface which may deviate a little from the actual geometry and introduces minor distortions not associated with any defect. Hence the displaced point cloud is considered the primary output and this surfaced STL is a visualization derived from that smoothing process. The reconstruction of the predicted branch follows a similar approach where the deviation predicted by the model is used, and the reconstruction output demonstrates the model’s learned mapping.

3.9 Defect Quantification

This stage summarizes each defect’s severity score as three geometric proxies computed from the signed deviation field over the defect region. A batch process runs over a part’s samples, reads the NPZ files for measured descriptors and applies the trained model for predicted descriptors. The output is in the form of a single CSV file containing both these descriptors per sample along with their difference.

The first geometric proxy is area, which represents the surface area of the defect:

$$A(\text{mm}^2) = N_r \times \frac{A_{\text{part}}}{N_{\text{total}}} \quad (3.5)$$

where N_r is the number of region points, A_{part} is the nominal part surface area, and N_{total} is the total number of sampled points. The term $A_{\text{part}}/N_{\text{total}}$ is the surface region represented by one sampled point which is valid because the nominal cloud

has uniform surface density due to area-weighted sampling, so multiplying the per-point patch by the number of region points gives the area of the defect.

The second is depth which is the mean value of the absolute deviation over the defect region.

$$\bar{d} = \frac{1}{N_r} \sum_{i=1}^{N_r} |d_i| \quad (3.6)$$

The maximum depth value is bounded by the deviation clamp $d_{\max}$ applied during data generation and reflects that clamp limit rather than the true maximum defect depth. Hence the mean depth is used as the primary depth measure. The 95th-percentile depth is also to some extent affected by the clamp at the extreme values. The third proxy measure is volume which refers to the net change in material:

$$V = \sum_{i=1}^{N_r} d_i \cdot \frac{A_{\text{part}}}{N_{\text{total}}} \quad (3.7)$$

summed over region points with signed deviation d_i . Sign follows the defect type and hence this proxy differentiates between material added and material removed defects.

All three descriptors are computed for both the measured and the predicted deviation, and the CSV in addition shows the difference in mean depth between the measured and predicted values for each sample which quantifies the model’s severity-prediction error. None of the three descriptors is a certified metrology measurement but rather severity proxies as they are affected by the limitations of pose registration and deviation measurement, and also area and volume are scaled according to the number of sample points and deviation clamp. Thus, these descriptors are valid for relative comparison between defects, between samples and between measured and predicted results, but not as absolute physical values. Due to unavailability of ground-truth 3D defective geometry in the dataset, validation of these proxies is not possible, and this is stated as a limitation.

4

Results

4.1 Anomaly Detection and Part Segmentation

4.1.1 Anomaly Detection

Five anomaly-detection methods across three modalities were tested using part 3M2C. Three methods were based on 2D RGB images, one was based on geometry and one was 2D-3D fusion. Each model was trained with 48 nominal part images and tested on 81 defective part images across five defect types, with image-level AUROC considered the primary metric. Table 4.1 shows the overall AUROC score for each model while Table 4.2 reports AUROC score for each defect. From observation, there was considerable variance in performance across the models and defect types. Among all the models, 2D image-based PatchCore produced the most credible/relevant AUROC result, although it was not the highest. A uniform score of 100% for PaDiM across all defect types indicates a likely evaluation-setup issue rather than genuine perfect separation, hence even this high-scoring method was not trusted. The overall anomaly detection results were not sufficiently trustworthy for further use in the pipeline. Together with the background-contamination issue described below, these results motivated the use of the dataset annotations.

Model	Modality	Image AUROC (%)	Inference Time (min)
PaDiM	2D RGB	100	0.4
CFA	2D RGB	98.5	1.9
PatchCore	2D RGB	56.6	0.7
3D PatchCore	3D Depth	46.4	2.2
M3DM	2D-3D Fusion	50.4	9.6

Table 4.1: Image-level AUROC comparison across five anomaly detection models on part 3M2C.

4.1.2 SAM Part Segmentation

Following the anomaly-detection results, it was noticed that the background elements in RGB images affected the anomaly scores mentioned in Section 4.1.1. To isolate the actual part from those background elements, the Segment Anything Model (SAM) with ViT-B backbone mentioned earlier in Section 2.4 was utilized to generate a part mask for each image. Figure 4.1 illustrates the SAM output for

4. Results

Defect Type	CFA	PaDiM	PatchCore	3D PatchCore	M3DM
Bulge	99.7	100.0	64.6	41.4	54.6
Cut	100.0	100.0	60.4	84.4	94.7
Hole	100.0	100.0	83.1	40.2	42.1
Over-Extrusion	93.5	100.0	10.8	37.5	9.4
Under-Extrusion	99.0	100.0	63.9	29.4	51.1

Table 4.2: Per-defect image-level AUROC (%) for each model on part 3M2C.

3M2C from three viewpoints for comparison: the original RGB image, the SAM binary mask and the mask boundary overlaid on the RGB image. In this sample, the part is almost cleanly isolated from the background but this sample also shows the role of lighting, background and part placement in segmenting the part from the background.

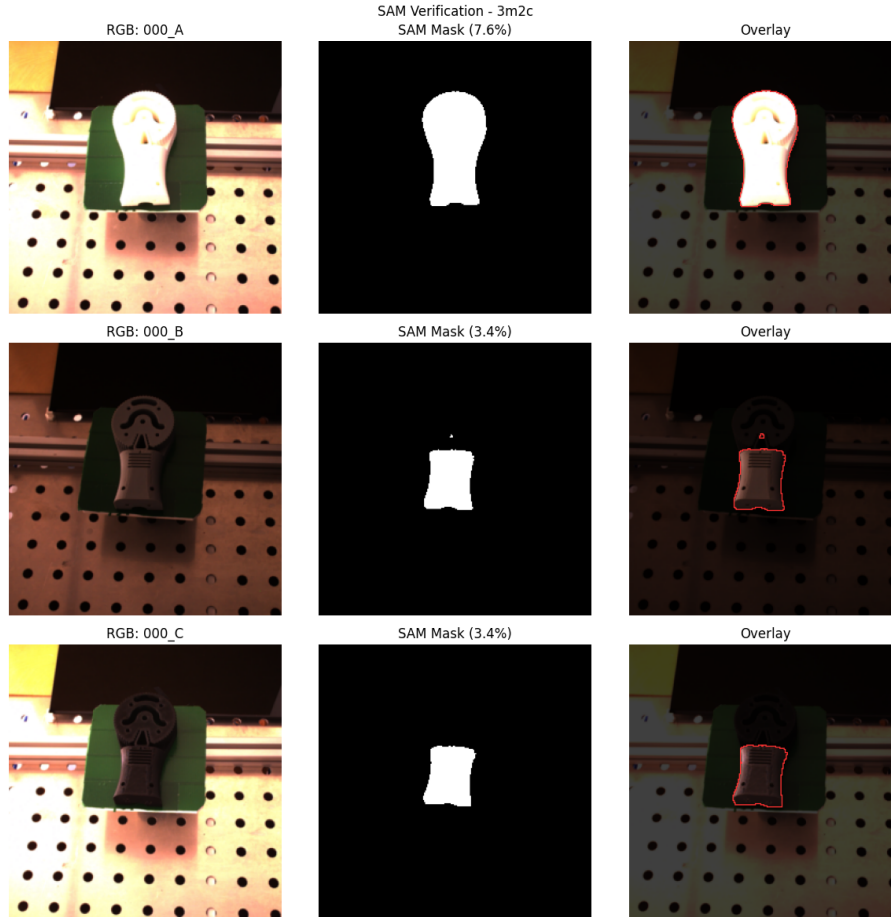


Figure 4.1: SAM segmentation representation for 3M2C with three viewpoints: RGB image, binary SAM mask, Overlay with mask boundary on RGB image

SAM generated a potential part mask for every image in the dataset. But after qualitative inspection it was observed that not every selected mask was correct. For some samples the selected mask also captured background elements as part of the actual component or excluded regions of the actual part, as is also visible in the

figure. Since the annotations provided in the dataset were used in the pipeline, the accuracy of mask generation was not quantified and the method is reported here for completeness.

4.2 Defect-to-Surface Mapping

The verification of the 2D defect to 3D surface mapping is reported here with the metrics defined in Section 3.5.2. As mentioned in Section 3.5.2, the round-trip IoU results coincide with the valid fraction, hence the results reported here in Figure 4.2 are in terms of on-part fraction and valid fraction.

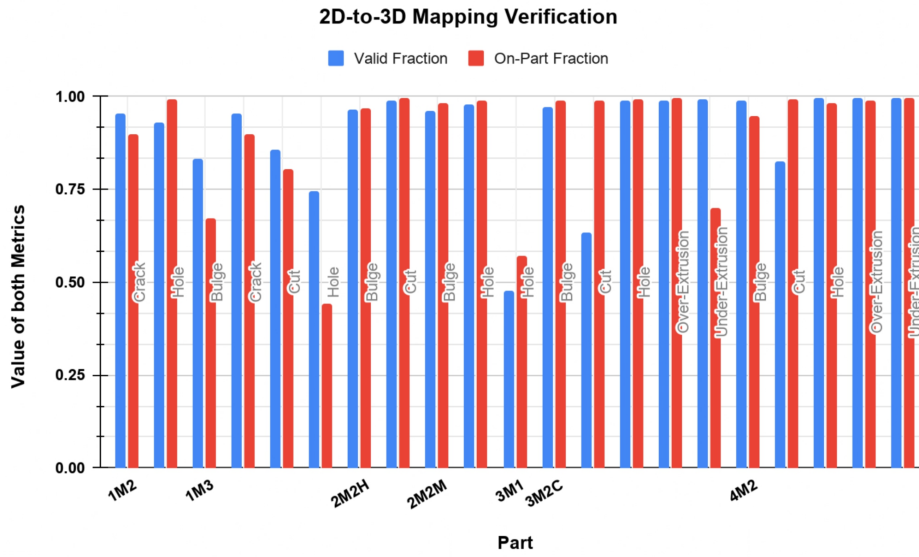


Figure 4.2: 2D-to-3D Mapping verification with on-part fraction and valid fraction

For parts with well-defined surface geometry, the mapping achieved high scores on both measures, indicating correct transfer of defect annotations to 3D surface. The on-part fraction, which is independently derived from an annotation not used during mapping, provides supporting evidence because a high value indicates the right correspondence of mapping. If the part’s defect points are displaced by any crop or offset error, the value of this on-part measure collapses. For parts with missing-material defects, such as holes on 3M1 and 1M3, very few mask pixels may map to captured surface points; consequently, the on-part measure is low. This depicts the nature of missing-material defects producing dropout, not an error in the mapping process.

For a few parts such as 1M3, the overlay visualization seems vertically inverted. This is not a mapping error but results from pose-mirror ambiguity or a pose-estimation error. The 2D to 3D correspondence is correct, but orientation of visualization is based on the pose estimation, so that error propagates downstream. For some parts, such as 3M1, which is a small part with very few defects, some of the overlay markings fall near or outside the part boundary and the overlay is less clear at

that scale. In such cases, the on-part fraction would be the appropriate measure to estimate the mapping quality.

4.3 Pose Estimation / Alignment

The quality of the registration between the PLY scan of each part and its nominal CAD is reported in this section. This quality determines the defect placement on the reconstructed geometry and it is evaluated using two indicators because no ground-truth pose is available in the dataset. These two proxy indicators, defined in Section 3.6.4, are pose-fit, the fraction of scan points within 2 mm of the CAD after alignment and on-part fraction, the fraction of defect points that lie within the region radius of the CAD after registration. These two indicators are supplemented by the median distance between scan points and the surface. The fidelity in this work refers to the average registration quality for a part, and parts are classified as good-fidelity and low-fidelity based on the values of registration quality metrics for each part. Fidelity is averaged for each part and does not guarantee the reconstruction of any particular sample is correct.

Part	Mean pose-fit	Mean on-part	Median residual (mm)	Fidelity
3M2C	0.46	0.678	-	Good
1M2	0.642	0.793	1.49	Good
1M3	0.614	0.619	1.49	Good
4M2	0.559	0.733	1.77	Good
2M2H	0.324	0.466	3.32	Low
2M2M	0.317	0.513	3.39	Low
3M1	0.574	0.633	1.57	Good

Table 4.3: Mean per-part pose-estimation metrics for accepted samples

Table 4.3 shows the score achieved by each part across each registration quality metric. These parts are categorized into good- and low-fidelity groups. The parts showing pose-fit value above 0.45 and residual between 1.4-1.9 mm are considered good-fidelity parts for which the scanned face is well aligned with the CAD. The low-fidelity parts show pose-fit value around 0.32 and residuals above 3 mm because these parts have geometries that make single-face registration difficult, resulting in loose alignment and less precise defect placement. Part 3M1 moderately registers to CAD as per registration quality metrics and is therefore considered to have good registration fidelity but due to its few and small hole defect markings, the number of high-quality samples is less for this part. As discussed in Section 3.6.1, the front/back mirror ambiguity issue is not completely solved for parts with low-fidelity but was reduced after using pose selection based on relief-weights. In these parts, some samples are registered to a mirrored pose of the part which results in defect placement in the right region but on the opposite side or face of the part. This issue is directly observed in the reconstructed defect-aware geometry and is the main reason for the placement error. This error results from registering a single-view scan to the CAD without camera calibration information and not of the reconstruction.

4.4 Deviation Prediction

The two PointNet++-based models, the geometry-only model and the heatmap-enabled model, were trained under the conditions mentioned in Section 3.7. Training was performed on a laptop equipped with an AMD Ryzen 5 processor, 8 GB of RAM, and an NVIDIA GeForce GTX 1650 GPU, using CUDA for GPU acceleration. A total of 348 accepted samples pooled from all seven parts were used with an 80/20 split stratified by defect class and trained for 80 epochs. This section presents the deviation magnitude and sign prediction results and the comparison against the class-average baseline for each defect class.

Figure 4.3 shows the Heatmap-enabled model’s overall test Mean Absolute Error (MAE) over all test points at each epoch along with the class-average baseline line shown for reference. The MAE is weighted by the number of test points for each defect. As shown, the error drops sharply within the first 20 epochs from around 0.9 mm to 0.73 mm, and it stays below the baseline (1.16 mm) throughout all epochs. Although the error fluctuates slightly between epochs 30 and 50 which do not indicate instability but are within the expected variation on a small test set, the error remains essentially flat from epoch 20 onward. The final epoch 80 is reported for all results.

Heatmap-enabled Model Test MAE at every Epoch

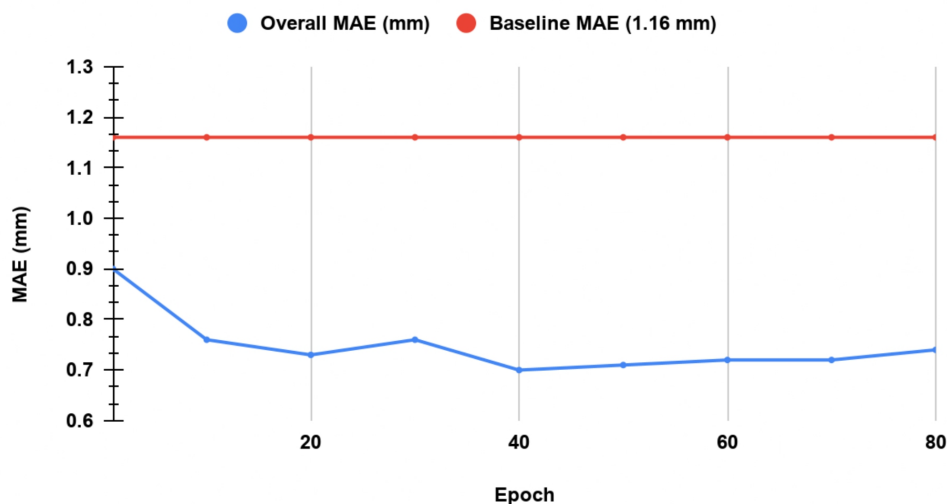


Figure 4.3: Test MAE at every Epoch for Heatmap model

Figure 4.4 illustrates the training and validation loss across the 80 epochs for the heatmap-enabled model. The training loss decreases steadily throughout training, while the validation loss decreases over the first 40 to 50 epochs and then stabilizes. It is important to note that the validation loss remains close to the training loss even when the training continues and both converge to similar values by the final epoch. This indicates that the model generalizes to unseen validation samples instead of overfitting the training data. This suggests that the model learned a relationship

rather than merely memorizing the training data.

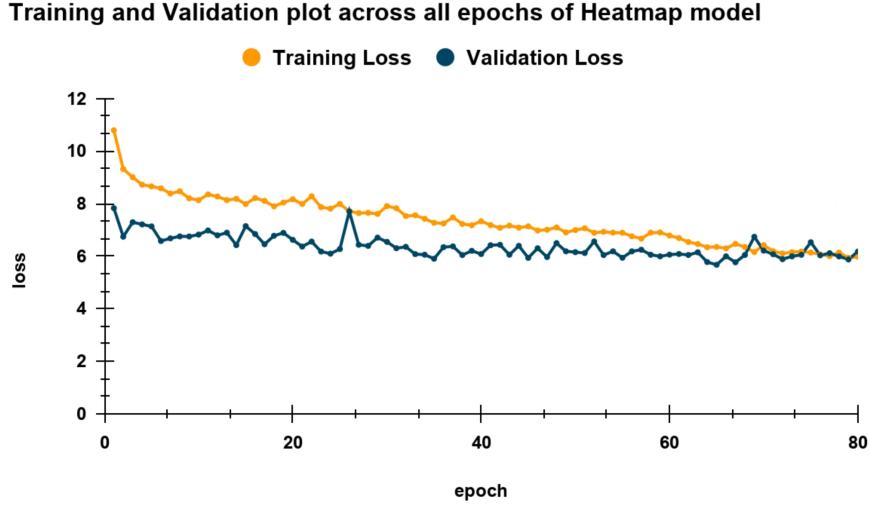


Figure 4.4: Training and Validation Loss plot across all epochs of Heatmap Model

4.4.1 Magnitude

Both Geometry-only and Heatmap-enabled learned models outperformed the overall class-average baseline as shown in Figure 4.5, which indicates that the location of the defect region itself also carries some information about defect magnitude. The heatmap-enabled model even shows higher performance and reduces the overall error by approximately 28% relative to the geometry-only model consistently for every defect type. The geometry-only model failed to outperform the baseline only for under-extrusion defect (1.75 vs 1.643) with only 21 samples and the largest deviations. For the same defect the heatmap-enabled model reduces this MAE to 1.226.

4.4.2 Sign Accuracy

Figure 4.6 shows a comparison between the sign accuracy predicted by the geometry-only model and the heatmap-enabled model. The geometry-only model is trained to directly predict the signed deviation similarly to the heatmap-enabled model, but without the heatmap input channels; it is distinct from the magnitude-only formulation described in Section 3.7. This helped in isolating the heatmap effect, as sign prediction without heatmap gives unstable and near random sign accuracy, while the heatmap-enabled model recovers it. However, the magnitude-only formulation which directly assigns sign from defect class, is not included in this comparison as sign accuracy from that formulation would be evidently perfect. There is a clear visible difference between the two models' sign-recovery performance from Figure 4.6. The sign accuracy of the geometry-only model is low and fluctuates throughout the training process without converging (for example sign accuracy of bulge defect

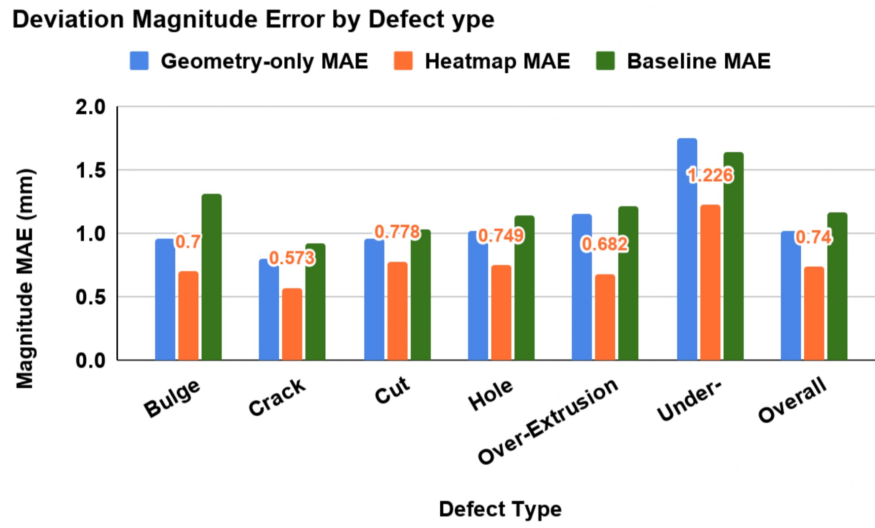


Figure 4.5: Deviation magnitude prediction comparison of both models with baseline

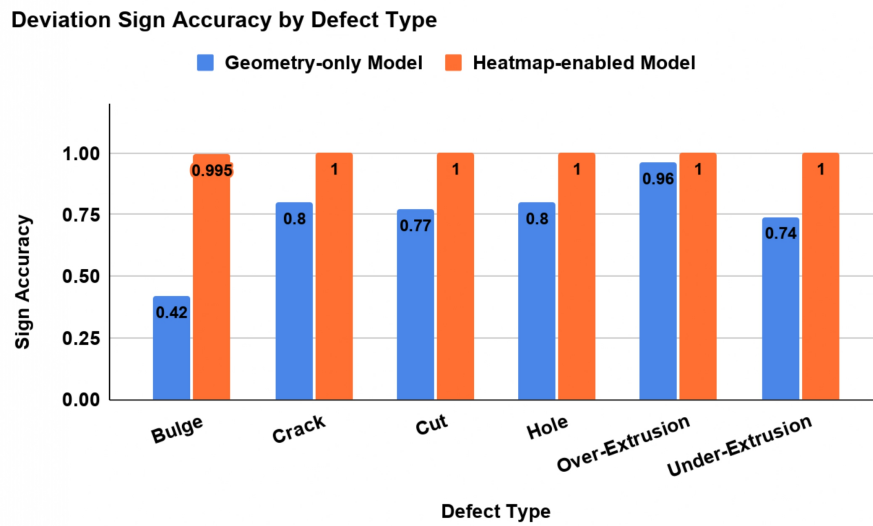


Figure 4.6: Deviation prediction sign accuracy comparison of both models with baseline

ranges from 0.00 to 0.69 across epochs), indicating the absence of direction information in the input; instead, the direction is guessed by the model. By contrast, in the heatmap-enabled model, sign accuracy remains stable at 0.99-1.00 throughout training. This is one of the critical findings of this thesis which demonstrates that the direction of deviation is recoverable when it is encoded in the input, such as the heatmap sign channel, but cannot be recovered from nominal geometry alone. This high sign accuracy of the heatmap-enabled model is the expected outcome of the presence of the sign information in the input and not evidence of the model's inference regarding direction. Thus, these results convey the exact information required to be supplied in order to achieve a scan-free pipeline.

4.4.3 Interpretation

The heatmap-enabled model reduced the magnitude error relative to both the geometry-only model and the baseline for each defect class. Also, this heatmap-enabled model recovered the deviation sign on significant points whereas the geometry-only model was unable to do so. These results indicate that the information about the deviation magnitude and direction can be provided with a rough 2D heatmap but not from nominal CAD geometry alone. Currently, the heatmap is created from the measured deviation from the scan but creating it instead from an RGB image without using a scan is treated as future work. The quick convergence of the heatmap-enabled model indicates that the model learned only a shallow refinement of an already informative input instead of performing deep geometric reasoning which is further examined in the Discussion.

4.5 Geometry Reconstruction

The results of the core part of this thesis, reconstruction of defect-aware geometry are presented in this section. The reconstruction stage for each sample produced two forms of output: the primary output is a displaced point cloud in PLY form and the other is a deviation map in VTK form. The reconstruction is performed using the measured and predicted deviation separately. The reconstructed parts shown below are at actual displacement scale unless an exaggeration factor for visualization is specifically mentioned. Figure 4.7 and Figure 4.8 illustrate the output forms of the geometry reconstruction stage for sample 006 of part 3M2C with bulge defect. In the displaced point cloud shown in Figure 4.7(a), the defect direction dictates the color of the defect region where red represents a material-added defect whereas blue represents material removed defects. In this sample the defect region points are physically raised above the nominal surface points along their normals representing a bulge defect. The deviation map in Figure 4.7(b) shows the signed deviation as a scalar field color-mapped on the nominal part surface. These two are the original forms of output of this stage. Figure 4.8(a) shows the watertight surface created from applying Screened Poisson surface reconstruction to the deviated point cloud, where the defect is visible as a raised region colored according to the type of defect. Figure 4.8(b) shows the exported STL after surface reconstruction for the purpose of CAD inspection. Together, these forms of output show the defect as reconstructed

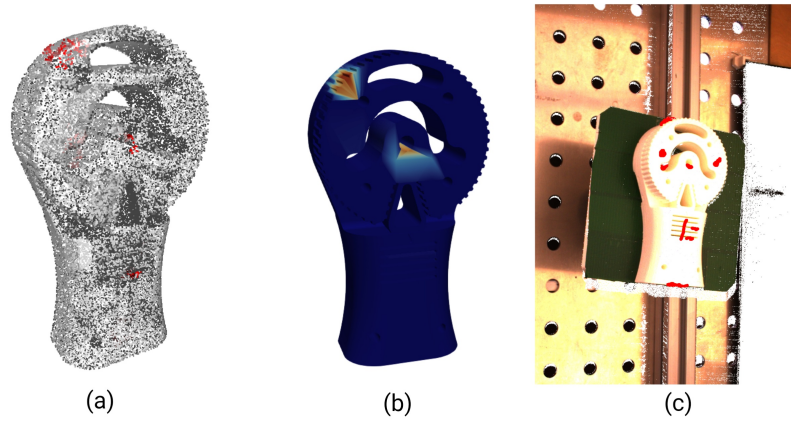


Figure 4.7: Output forms of Reconstruction of 006 sample of 3M2C part with Bulge defect in (a) Deviated Point Cloud (b) VTK Deviation Color Map along with (c) Overlay image

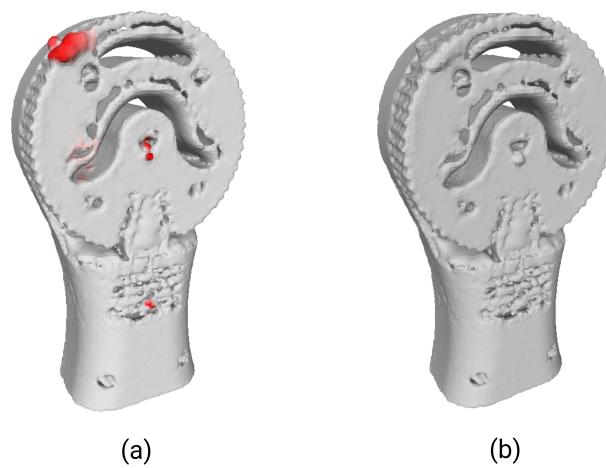


Figure 4.8: Derived Output forms of Reconstruction of 006 sample of 3M2C part with Bulge defect in (a) Screened Poisson Surface Reconstruction (b) Exported STL after Surface Reconstruction

geometry as well as a quantitative deviation field.

In the above sample, the bulge is mapped on the upper surface of the part and the elevated region corresponds to the annotated defect area, along with magnitude of displacement consistent with the measured deviation. Two visualization characteristics are observed here. The deviation map in Figure 4.7(b) illustrates the signed deviation colored on the nominal surface. Although the map contains no geometric deformation, the colors have a faceted appearance within the defect area due to linear interpolation of the deviation across large, uneven triangles of the nominal mesh. This visualization artifact appears due to mesh resolution and it does not impact the deviation values as the location of the defect and its severity are still visible by the color intensity. Second, the exported STL in Figure 4.8(b) shows minor surface irregularities or distortions in some regions of the defect such as near boundaries or thin features. These are caused by the screened Poisson surface reconstruction method fitting a smooth surface across a few or irregularly oriented nominal points. The displaced point cloud does not show any of the above characteristics or artifacts and hence is considered the primary output while the surfaced STL is considered as a CAD-compatible representation.

4.5.1 Reconstruction for Each Defect Type

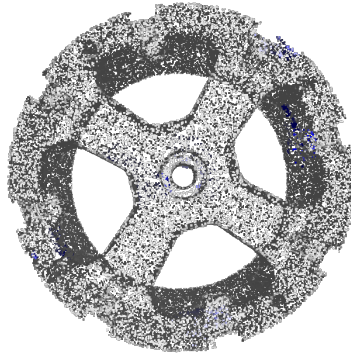


Figure 4.9: Deviated Point Cloud of sample 009 of part 1M2 with Crack defect

Figures 4.9-4.15 illustrate the displaced point cloud for one representative sample of each defect type compared against their respective reference, either the RGB image and annotation or the projected overlay, whichever option enables the clearer comparison for that sample given part color and lighting condition is considered. The reference shown alongside each defect type allows the spatial extent of the displacement to be compared with the annotation and helps identify placement offsets. Each defect is nearly localized to the correct region of the part along with proper direction although a few placement offsets within the region are visible due to the registration precision mentioned in Section 4.3. The small defects such as cut and hole are less clearly visible than the large defects such as over and under extrusion

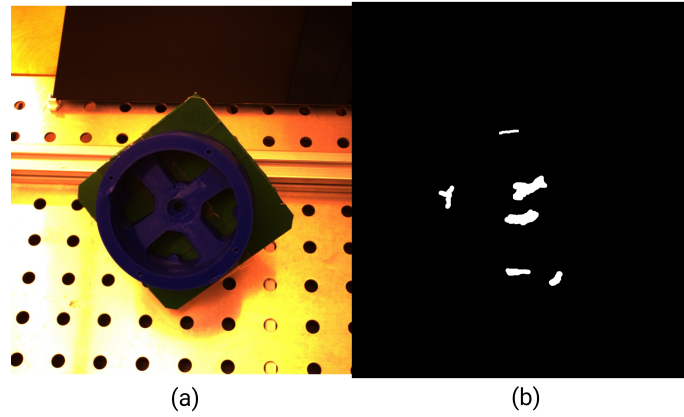


Figure 4.10: Sample 009 of part 1M2 with Crack Defect Reference from Dataset
(a) RGB Image (b) Ground-Truth Annotation



Figure 4.11: Deviated Point Cloud of sample 011 of part 3M2C with Cut defect

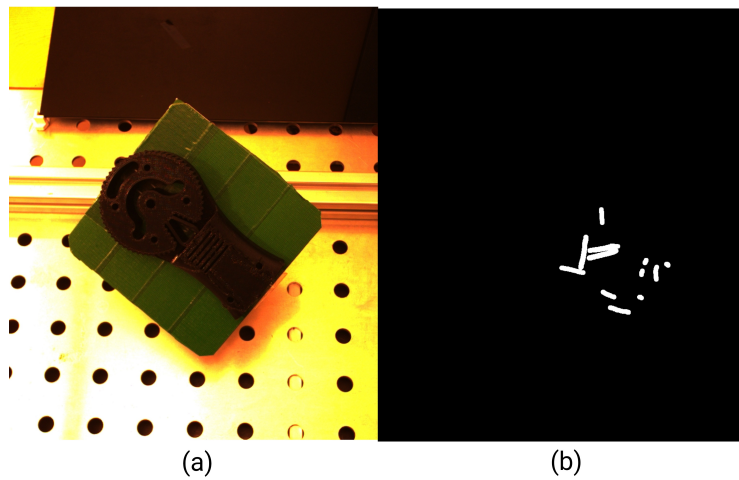


Figure 4.12: Sample 011 of part 3M2C with Cut Defect Reference from Dataset
(a) RGB Image (b) Ground-Truth Annotation

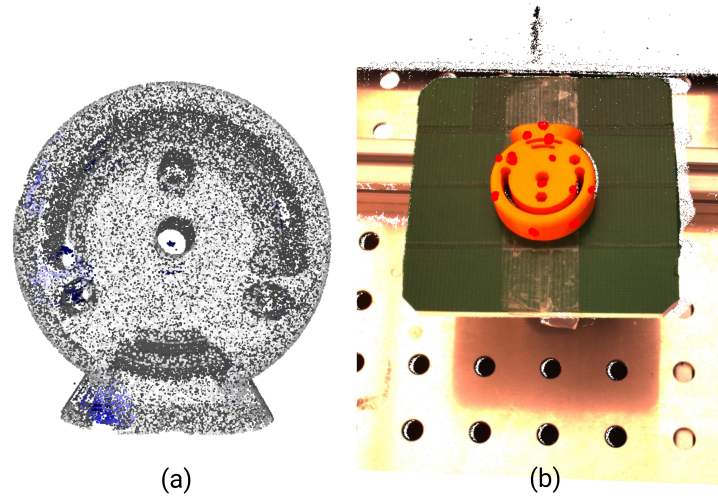


Figure 4.13: Sample 004 of part 4M2 with Hole Defect (a) Deviated Point Cloud (b) Overlay Image

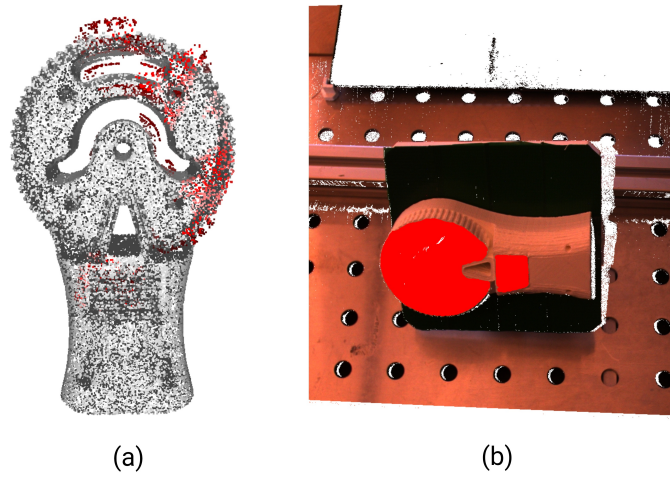


Figure 4.14: Sample 014 of part 3M2C with Over-Extrusion Defect (a) Deviated Point Cloud (b) Overlay Image

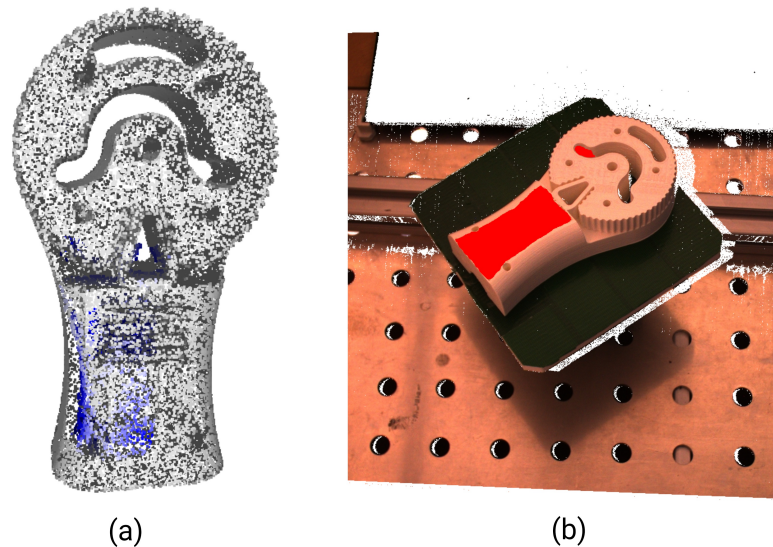


Figure 4.15: Sample 001 of part 3M2C with Under-Extrusion Defect (a) Deviated Point Cloud (b) Overlay Image

at the scale of the whole part; hence another view focusing specifically on the defect region of part is included alongside the full point cloud.

The markings of crack defect from Figure 4.10 are visible along the part surface in the deviated point cloud shown in Figure 4.9. Due to the round shape of part 1M2, a few markings that map onto the curved side of the part cannot be captured in a single view but the visible defect markings correspond to the ground-truth annotation and a zoomed view of the defect region is illustrated in Figure A.1.

In Figure 4.11, many markings of the cut defect on part 3M2C correspond to the reference annotation from Figure 4.12 but those markings may not be fully visible at the scale of the whole part hence a view of the same sample focusing on the defect region is illustrated in Figure A.2.

The hole defect appears as many small, scattered dots, showing the discrete nature of the defect. Through observations made across all parts with this defect, not all hole markings appear in the reconstruction as visible in the deviated point cloud in Figure 4.13 of part 4M2. A defect involving little missing material captures only a small number of surface points, hence the reconstruction recreates the scattered character of defect but does not reproduce every hole annotation.

Figure 4.14(b) shows that the over-extrusion defect occupies a large portion of the part surface, and so the defect is clearly visible in the point cloud in Figure 4.14(a) of part 3M2C. The reconstructed defect region corresponds with the annotation, and the offset in defect placement is consistent with the precision of registration.

The under-extrusion defect also covers a considerable portion of the part surface and is also clearly visible. Figure 4.15(b) shows defect projected onto the 3M2C part overlay, the location of the defect is in the central portion of the part's grip structure, but as per Figure 4.15(a) the defect is shifted toward one side in reconstructed geometry. This offset in defect localization occurred because of registration limitation rather than any error in reconstruction, though the offset here is modest.

In each of the above cases, the defect-aware geometry is reconstructed with the correct direction and a spatial extent related to the defect annotation. The placement offsets are bounded by the single-view registration precision mentioned in Section 4.3 and the mirror ambiguity issue discussed below.

4.5.2 Measured vs Predicted

Figure 4.16 shows the comparison between the reconstructed geometry of sample 006, containing a bulge defect on part 3M2C using measured and predicted deviation. The reconstructed geometry produced using the predicted deviation in Figure 4.16(b) correctly shows the defect at its location and intended direction; also the predicted defect magnitude agrees with the measured deviation within the characterized error. However, this will not be the case for every sample, as the quality of predicted reconstruction depends on the prediction accuracy of the model for each sample and the model’s prediction error varies by sample and defect type. This predicted reconstruction is not a validated output but rather it demonstrates the ability of the pipeline to reconstruct defect-aware geometry using the output of the learned model.

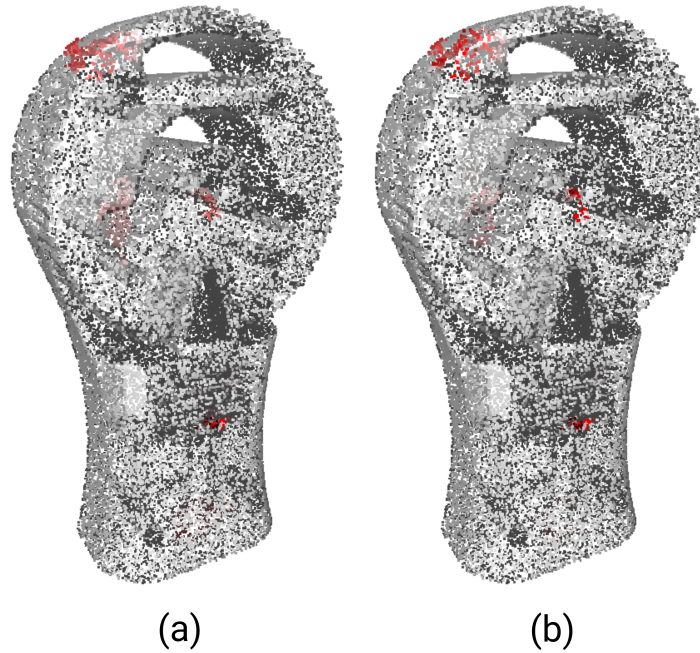


Figure 4.16: Deviated Point Cloud of sample 006 of part 3M2C with bulge defect with (a) Measured Deviation (b) Predicted Deviation

Table 4.4 shows the prediction error of the heatmap-enabled model, calculated as the difference between the predicted and measured mean depth for each sample and grouped by defect type across all seven parts. The mean error for most defect classes is small, at around 0.1-0.3 mm, which indicates that the model is fairly accurate in reproducing defect depth. The over-extrusion defect shows the highest mean er-

ror and a significantly larger standard deviation, which is in line with the broad, low-relief defect whose depth is most affected by the high-pass filter mentioned in Section 3.6.

Defect	Mean Depth Error (mm)	Std Deviation	Samples
Bulge	0.185	0.155	89
Crack	0.237	0.216	48
Cut	0.16	0.18	75
Hole	0.192	0.17	87
Over-Extrusion	0.585	0.258	28
Under-Extrusion	0.365	0.291	21

Table 4.4: Prediction Error and Standard Deviation per Defect for Heatmap-enabled Model

4.5.3 Mirror Ambiguity and Fidelity Exception

The part-level fidelity classification in Table 4.3 indicates the overall quality of registration for each part; however, it does not reliably assess the accuracy of the reconstruction of an individual sample which is illustrated by the following two observations. The placement offsets observed for the representative samples of each defect type above are evaluated here in the two most relevant cases: mirror ambiguity and the divergence between the registration-quality score and reconstruction accuracy. The first observation is that even with high registration quality scores, the reconstruction of some samples is unsatisfactory such as sample 002 of 4M2 part with under-extrusion defect. This sample achieved a high registration score, including a pose-fit of 0.70 and on-part fraction of 0.85 but in the reconstructed geometry the defect is placed on the wrong side of the part due to a left-right mirror ambiguity visible in Figure 4.17. The second observation was that even with low fidelity and low registration scores, some samples were reconstructed much better than expected, such as sample 014 of 2M2M part with bulge defect. Although this sample achieved low pose-fit score of 0.32, the defect is placed on the correct side of the part in reconstructed geometry along with many defect markings corresponding to the overlay image of the sample visible in Figure 4.18.

The above-mentioned samples are not unique as this variation was commonly observed across the evaluated samples; a high registration quality score does not guarantee accurate placement while a low registration quality score does not ensure a reconstruction failure. The reason is that registration quality metrics assess the overall registration plausibility, but the correct placement of the defect also depends on the front/back and left/right mirror ambiguity which is not completely mitigated due to the single-view registration regardless of the overall fit. The reconstruction correctly produces the provided deviation and its location but the placement error occurs in the upstream registration.

Accordingly, the results of reconstruction are reported with explicit acknowledgment of this limitation. When the registration works correctly, the reconstruction recreates the defect in the correct position with the correct direction and a severity

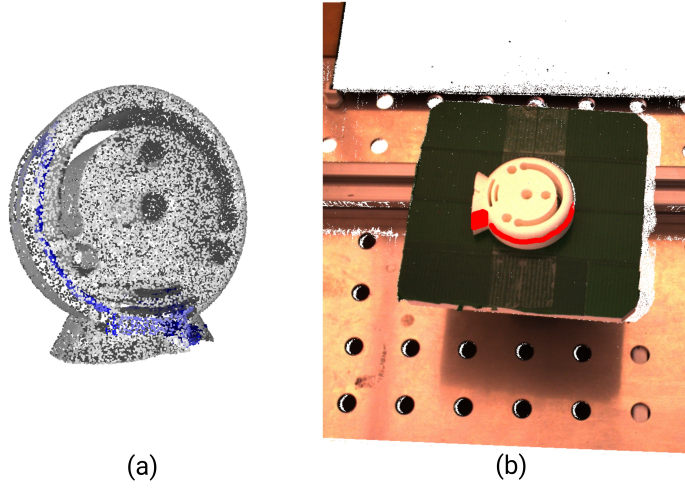


Figure 4.17: Mirror Ambiguity in sample 002 of part 4M2 with Under-Extrusion defect (a) Deviated Point Cloud (b) Overlay image

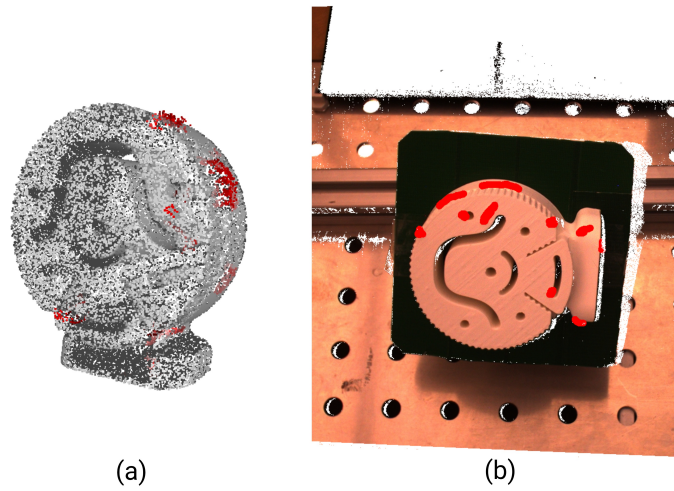


Figure 4.18: Fidelity Exception in sample 014 of part 2M2M with Bulge defect (a) Deviated Point Cloud (b) Overlay image

corresponding to the measured deviation as shown in the representative examples. But when the registration introduces a mirror-ambiguity problem, the defect appears on the wrong face or side of the part but mostly its shape and severity remain correct. The single-view registration without camera calibration constrains this limitation of placement and the accuracy of reconstruction which is further discussed in the limitations section.

4.5.4 Summary of Geometry Reconstruction Results

The reconstruction stage produced defect-aware geometry for all seven parts, in which defects were localized to the correct region of the part with the correct direction and severity as indicated by the measured deviation. The precision of defect placement on the part is bounded by the limitations of single-view registration, which introduces a mirror ambiguity which is not completely mitigated and is the main limitation on the accuracy of the reconstruction. Among all the forms of output, the deviation map shows a color artifact based on mesh resolution and the exported STL after Screened Poisson surface reconstruction inherits minor smoothing; the deviated point cloud format properly shows the defect geometry without exhibiting the mesh-interpolation faceting or Screened Poisson smoothing artifacts mentioned above and hence it is the primary output of the pipeline. The reconstructed geometry with predicted deviation demonstrates the possibility of producing geometry from the output of the learned model, with fidelity dictated by the model error characterized in Section 4.4.

4.6 Defect Quantification

The three proxy geometry descriptors, area, mean depth, and volume, which are computed over the defect region from both measured and predicted deviations for all seven parts are presented in this section. Table A.3 summarizes the average values of these descriptors for each defect for each part. Also, the mean depth prediction error for each defect type is reported in Figure 4.19 as the heatmap-enabled model’s accuracy.

For all parts, there is no sign error in either the measured or predicted volume which confirms the correct sign convention propagation through the reconstruction and quantification stages provided by the heatmap with positive sign conveyed for material added defects and negative sign for material removed defects. The difference between measured and predicted mean depth is between 0.1-0.3 mm for most parts and defects despite varying part geometries and registration quality. This suggests the model’s accuracy in reproducing defect severity is stable across the part geometries evaluated and supports the idea of supplying a heatmap as input. The only exception is the over-extrusion defect with depth error between 0.49-0.71 mm because of its broad area and low relief. Also, the depth of this over-extrusion defect became hardest to estimate because its severity is partially suppressed by the high-pass filter.

These three descriptors are not validated measurements; rather, they are severity proxies. Area and volume vary according to deviation clamp and density of sam-

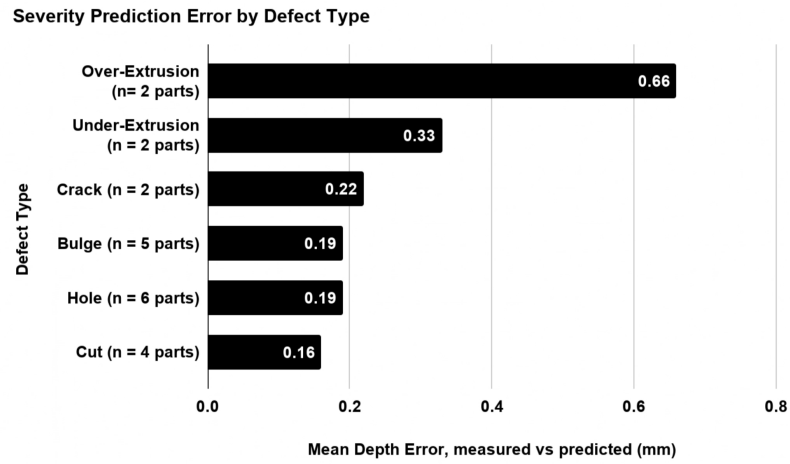


Figure 4.19: Mean depth prediction error for each defect type by heatmap-enabled model

pled points, thus their absolute magnitudes are valid only for relative comparison between defects, samples and between the measured and predicted branches. For example, the large area of over-extrusion defect actually reflects its broad coverage rather than inflation of the measured value. The validation of these three proxies with an actual physical reference is not possible due to the absence of ground-truth 3D defect geometry in the dataset, and this is stated as a limitation.

5

Discussion

5.1 Interpretation of the Results

The findings of this work do not set a performance benchmark but rather explain the information required for defect reconstruction. The comparison between the geometry-only and heatmap-enabled models clearly indicates that nominal CAD geometry remains identical for any defect so it alone cannot be used to recover direction of deviation and magnitude is weakly encoded in the defect’s location on the part. The geometry-only model modestly outperforms the baseline but predicts deviation magnitude only weakly and cannot predict the sign of the deviation. But providing information about deviation through a coarse 2D heatmap allows recovery of both magnitude and direction of the defect, where magnitude recovery shows around 28% lower error than the geometry-only model and 36% lower than the class-average baseline.

Hence, the information content of the input is a more important constraint than the model’s capacity. This observation is supported by the rapid convergence of both models, which reach near-final performance within twenty epochs; this suggests shallow learning by the network rather than deep geometric structure extraction. The reason behind the better performance of the heatmap-enabled model lies in its input, as it contains the deviation information in coarse form, and the heatmap-enabled model further refines it onto the CAD surface. This insight rephrases the problem of reconstruction as now the crucial requirement is not to build a more powerful predictor network but to find a way to provide the required deviation information during inference without using a 3D scan.

5.2 The Scan-free Question

The main aim of this work was to move toward defect reconstruction from a 2D image without a 3D scan at inference. This, however, was not completely achieved and it is important to discuss why. The current pipeline depends on scans at several points: the 2D-to-3D mapping reads 3D coordinates from the sensor’s organized point cloud, the deviation is measured by comparing two scans, and the heatmap that enables sign and magnitude recovery is itself derived from that measured deviation. Each of these points serves as a source of information that cannot be derived from a 2D image alone. Nevertheless, the heatmap approach shows the path by providing a coarse anomaly-style severity-and-sign map and not exact deviation; this input is presented in a form that an RGB-based two-dimensional detector could reproduce,

at which point no scan would be required at inference. The remaining obstacle of image-to-CAD registration without camera calibration is discussed in the limitations below. This work contributes and establishes through the geometry-only vs heatmap comparison what a future scan-free system must supply, and demonstrates that a coarse two-dimensional input is sufficient.

5.3 Design Choices and Trade-offs

Some choices in the methodology of this thesis involved trade-offs which shaped the current results.

In order to remove the print-to-print distortion from the measurement of deviation, introducing the high-pass baseline subtraction was required; otherwise, a smooth global offset would have been mistakenly considered as a defect. But on flat, low-relief defects like over-extrusion, the defect surface itself is involved in local baseline computation and hence this filter suppresses the actual defect signal. This is the main reason why over-extrusion is the most difficult defect to compute deviation. A defect-size-aware baseline, obtained only from defect-free surface, could solve this situation but it is outside the current scope.

To avoid the possibility of the model reproducing the original input, the coarse heatmap was created by binning, smoothing and lowering resolution. This allows the model to perform genuine reconstruction but at the same time the coarse heatmap enables the model to learn only a shallow refinement rather than a deep structure and questions the actual learning component. Despite this limitation, the coarse heatmap is the right choice here because a high-resolution alternative input would be impossible to obtain without a scan and would result in a circular formulation. For every reconstruction, the single-view registration creates barriers to accurate placement. Registering a single partial face to a complete 3D geometry is under-determined and introduces a front/back mirror ambiguity which is partially mitigated but not completely eliminated by pose selection based on the relief-weighted method. Multi-view scanning would help in this situation, but that is not possible because the dataset provides only single-view scans.

The reconstruction aimed to show defects as physical surface displacement by displacing each point along its normal. This displaced point cloud is without any deformation artifacts, but surface reconstruction into a mesh via screened Poisson introduces minor smoothing in areas with few points away from the defect. Hence the point cloud is the primary output and the reconstructed mesh is derived for visualization. Alternatively, directly deforming the nominal mesh was not pursued as it produced triangulation artifacts at the defect, while the point-cloud approach is used even though this requires a separate surfacing step for CAD-compatible output.

5.4 Limitations

5.4.1 Limitations Mitigated

1. Fake defect signals were generated due to global differences between two prints of a part. This issue was mitigated with a high-pass filter that performed local baseline subtraction and this filter was further used for all seven parts in the pipeline.
2. Initially the sign accuracy was limited to around 0.57 after applying the high-pass filter because the near-zero points diluted it. But it was overcome by computing sign accuracy only on meaningful points with deviation magnitude above 0.3 mm; this change increased sign accuracy to approximately 1.0 on meaningful points.
3. Parts having near-planar faces that fit either way show front/back mirror ambiguity issue. This ambiguity was partially reduced with relief-weighted pose scoring, which considers the distance between the mid-plane and elevated features that appear on only one face of the part as crucial information.
4. The geometry-only model weakly predicted deviation magnitude and sign due to missing information in the input, which was overcome by producing and providing a coarse heatmap encoded with defect direction, which improved the deviation magnitude prediction and recovered defect sign.

5.4.2 Limitations Dictating Thesis Work

1. In some reconstructed geometries, the defect lands on the wrong face or side of the part which is reduced but not completely mitigated as defect placement depends on registering the scan to the nominal CAD using only a single-view scan.
2. Over-extrusion is a defect with broad coverage and the high-pass filter flattens it more than required hence this defect shows the largest prediction error among the defects.
3. The ground-truth provided in the dataset consists of 2D binary images but no 3D ground-truth geometry; hence, without proper validation, the quantification descriptors are proxies and not actual measurements.
4. The dataset provides around 50 samples for each part which are sufficient for demonstration but not enough for training a strong learned model and it is indicated by rapid convergence in the training logs.
5. The train/test split is by individual samples stratified by defect type, so a test set exists but no entire part geometry is withheld; hence generalization to unseen geometry is untested and the documented errors are optimistic in that respect.

5.4.3 Limitations Out of Scope

1. This dataset does not provide any camera calibration and intrinsics data; hence, it is not possible to perform 2D-to-3D registration without using scans.

2. The heatmaps are derived from scans, and for a true scan-free pipeline, these heatmaps should be produced using an RGB-based 2D anomaly detector. This thesis tried anomaly detection but the results were not satisfactory and as the core focus of the thesis was 2D-to-3D and defect-aware geometry reconstruction, that approach was not further considered.
3. The RGB fusion, in which the shadows and surface texture provide key information is an appropriate approach for 2D-to-3D projection without using scan. But this approach is limited because the RGB images in the dataset were captured in inconsistent lighting. Although the 3D-ADAM paper states that the dataset was captured under controlled lighting, manual evaluation of various RGB images in the dataset revealed this limitation.

5.5 Future Work

The most immediate future work is to close the scan-free loop by creating the heatmap from RGB-based anomaly detection rather than from the measured deviation, eliminating the scan from inference for the prediction stage. This relies on image-to-CAD registration, which requires camera calibration parameters not provided in the dataset. Obtaining these is a prerequisite. The integration of photometric cues into the model, to recover sign and magnitude directly from the image, is a further possibility but requires controlled lighting, which the dataset does not provide. In addition to scan-free execution, employing multi-view scanning would resolve the mirror ambiguity and improve placement accuracy; furthermore, a defect-size-aware deviation baseline would improve broad-defect capture; and integrated anomaly detection and segmentation would replace the dataset-provided masks, completing the pipeline end-to-end. Finally, scaling to more parts would help to further test generalization.

6

Conclusion

This thesis created and tested a pipeline for projecting 2D defect annotations onto nominal 3D geometry and reconstructing defect-aware 3D geometry with the help of structured-light scan data provided in the 3D-ADAM dataset. The pipeline maps 2D defect masks to the coordinates of the 3D surface, registers every PLY scan to the nominal CAD, computes the signed deviation between defective and nominal scans, reconstructs the defect-aware geometry in the form of a displaced point cloud and deviation map, and finally quantifies the defect with the help of geometric descriptors.

The first research question asked whether such a pipeline could be created and the thesis demonstrated its feasibility. The 2D-to-3D defect mapping was quantitatively validated against an independent annotation, and the reconstruction stage created defect-aware geometry that localized defects in approximately the correct regions with the correct direction. The key limitation in this thesis is placement accuracy dictated by the single-view registration, which brings a mirror ambiguity issue that was reduced but not eliminated.

The second research question asked how accurately defect deviation could be predicted on the nominal CAD, and what input information is required for pure scan-free inference. The thesis work indicated that nominal geometry alone is insufficient to recover direction of deviation because the nominal surface remains unchanged regardless of defect sign, and defect magnitude is weakly encoded in defect location. Providing a coarse 2D heatmap containing defect-severity and sign information recovers direction and also reduces magnitude error by around 28% relative to a geometry-only model and 36% relative to a baseline. The accuracy of deviation prediction is quantified through the magnitude MAE weighted by the number of test points, where the geometry-only model achieved an MAE of 1.02 mm while the heatmap-enabled model achieved 0.74 mm against a baseline MAE of 1.16 mm. The mean depth error recorded is between 0.1-0.3 mm against the measured depth of 0.66-1.68 mm. Hence the deviation predictability of the model is determined by the information encoded in the input rather than the model's capacity. The heatmaps used in this thesis are derived from measured deviation between defective and nominal scans from the dataset but for actual scan-free inference, the key requirement will be to generate those heatmaps using anomaly detection based on RGB images. Regarding the hypotheses stated in Section 1.3.1, the first two were confirmed correct. The nominal CAD worked as a trustworthy reference on which defects were projected (H1), and the organized point cloud of the sensor provided a direct 2D-to-3D correspondence without any additional reconstruction step (H2). The third hypothesis that a learning model could predict deviation of defect on nominal CAD

only from the defect region, allowing scan-free inference, was invalidated in its original form. The comparison between geometry-only and heatmap-enabled model demonstrated that both the defect direction and, to a large extent, its magnitude cannot be recovered from the nominal geometry and the defect region alone, due to the fact that nominal geometry does not contain any information about material added or removed by a defect. This falsification itself is a key finding, as it indicates that for scan-free inference, the information about deviation should be provided from the input, for example through an anomaly-detection heatmap which is a requirement for future scan-free operation.

This thesis work makes two contributions: first, a scan-based pipeline based on the dataset’s scans for defect mapping, geometry reconstruction and defect quantification; and second, identification of the information required to predict defect deviation on a nominal CAD and for scan-free inference. The main direction for future work is to obtain the heatmaps from RGB-based anomaly detection which, together with the availability of camera calibration for image-to-CAD registration, would enable the transition of the pipeline to actual scan-free inference.

Bibliography

- [1] Jonatan Berglund, Erik Lindskog, and Björn Johansson. On the trade-off between data density and data capture duration in 3d laser scanning for production system engineering. *Procedia CIRP*, 41:697–701, 2016. Research and Innovation in Manufacturing: Key Enabling Technologies for the Factories of the Future - Proceedings of the 48th CIRP Conference on Manufacturing Systems.
- [2] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Mvtec ad — a comprehensive real-world dataset for unsupervised anomaly detection. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9584–9592, 2019.
- [3] Paul Bergmann, Xin Jin, David Sattlegger, and Carsten Steger. The mvtec 3d-ad dataset for unsupervised 3d anomaly detection and localization. pages 202–213, 01 2022.
- [4] P.J. Besl and Neil D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.
- [5] Vivek Bhandarkar, Harshal Shahare, Anand Mall, and Puneet Tandon. An overview of traditional and advanced methods to detect part defects in additive manufacturing processes. *Journal of Intelligent Manufacturing*, 36:4411–4446, 09 2024.
- [6] Thomas Defard, Aleksandr Setkov, Angelique Loesch, and Romaric Audigier. Padim: a patch distribution modeling framework for anomaly detection and localization. *CoRR*, abs/2011.08785, 2020.
- [7] Haoqiang Fan, Hao Su, and Leonidas Guibas. A point set generation network for 3d object reconstruction from a single image. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2463–2471, 2017.
- [8] Haoxiang Guo, Shilin Liu, Hao Pan, Yang Liu, Xin Tong, and Baining Guo. Complexgen: Cad reconstruction by b-rep chain complex generation, 2022.
- [9] Qiang Huang, Yuanxiang Wang, Mingdong Lyu, and Weizhi Lin. Shape deviation generator—a convolution framework for learning and predicting 3-d printing shape accuracy. *IEEE Transactions on Automation Science and Engineering*, pages 1–15, 12 2019.

- [10] Yu Jin, Haitao Liao, and Harry Pierson. Automatic feature-based point cloud alignment and inspection. *Procedia Manufacturing*, 39:484–492, 2019. 25th International Conference on Production Research Manufacturing Innovation: Cyber Physical Manufacturing August 9-14, 2019 | Chicago, Illinois (USA).
- [11] Yang Jing, Shaobo Li, Zheng Wang, Hao Dong, Jun Wang, and Shihao Tang. Using deep learning to detect defects in manufacturing: A comprehensive survey and current challenges. *Materials*, 13:5755, 12 2020.
- [12] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything, 2023.
- [13] Juheon Lee, Lei Chen, Juan Carlos Catana, Hui Wang, and Jun Zeng. Graphcompnet: A position-aware model for predicting and compensating shape deviations in 3d printing. *IEEE Transactions on Automation Science and Engineering*, 23:870–881, 2026.
- [14] Kisun Lee, Youngjun Yoo, and Duhwan Mun. Point cloud registration using pca and hyperparameter optimization. *International Journal of Control, Automation, and Systems*, 24(4):892–908, 2026.
- [15] Sungwook Lee, Seunghyun Lee, and Byung Cheol Song. Cfa: Coupled-hypersphere-based feature adaptation for target-oriented anomaly localization. *IEEE Access*, 10:78446–78454, 2022.
- [16] Jiaqi Liu, Guoyang Xie, Ruitao Chen, Xinpeng Li, Jinbao Wang, Yong Liu, Chengjie Wang, and Feng Zheng. Real3d-ad: A dataset of point cloud anomaly detection, 2023.
- [17] Yujia Liu, Anton Obukhov, Jan Dirk Wegner, and Konrad Schindler. Point2cad: Reverse engineering cad models from 3d point clouds, 2023.
- [18] A. Nihil Mathew, V. Pandurangan, and N. Venkaiah. Physics-informed intelligent framework for geometric prediction and compensation in metal additive manufacturing. *Journal of Intelligent Manufacturing*, May 2026.
- [19] Paul McHard, Florent P. Audonnet, Oliver Summerell, Sebastian Andraos, Paul Henderson, and Gerardo Aragon-Camarasa. 3d-adam: A dataset for 3d anomaly detection in additive manufacturing, 2025.
- [20] No Kap Park and Suk In Yoo. Evaluation of tft-lcd defects based on human visual perception. *Displays*, 30(1):1–16, 2009.
- [21] Charles R. Qi, Li Yi, Hao Su, and Leonidas J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space, 2017.
- [22] Charles R. Qi, Hao Su, Mo Kaichun, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 77–85, 2017.

-
- [23] Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter Gehler. Towards total recall in industrial anomaly detection. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14298–14308, 2022.
 - [24] Roham Sadeghi Tabar, Lars Lindkvist, Kristina Wärmefjord, Pasquale Franciosa, Dariusz Ceglarek, and Rikard Söderberg. Enhancing welding geometric precision: Analyzing the impact of weld path directions, sequences and locating schemes on displacement. *Appl. Sci. (Basel)*, 14(23):11144, November 2024.
 - [25] Abdul Rahman Sani, Ali Zolfagharian, and Abbas Z. Kouzani. Artificial intelligence-augmented additive manufacturing: Insights on closed-loop 3d printing. *Advanced Intelligent Systems*, 6(10):2400102, 2024.
 - [26] Roham Sadeghi Tabar, Kristina Wärmefjord, Rikard Söderberg, and Lars Lindkvist. Efficient spot welding sequence optimization in a geometry assurance digital twin. *Journal of Mechanical Design*, 142(10):102001, 04 2020.
 - [27] Amir Tjolleng, Joonho Chang, Jangwoon Park, Wonsup Lee, Myeongsup Cha, Jongchul Park, and Kihyo Jung. Development of a human-friendly visual inspection method for painted vehicle bodies. *Applied Ergonomics*, 106:103911, 2023.
 - [28] Yue Wang, Jinlong Peng, Jiangning Zhang, Ran Yi, Yabiao Wang, and Chengjie Wang. Multimodal industrial anomaly detection via hybrid fusion. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8032–8041, 2023.
 - [29] Tong Wu, Liang Pan, Junzhe Zhang, Tai Wang, Ziwei Liu, and Dahua Lin. Density-aware chamfer distance as a comprehensive metric for point cloud completion, 2021.
 - [30] Yang You, Mikaela Angelina Uy, Jiaqi Han, Rahul Thomas, Haotong Zhang, Yi Du, Hansheng Chen, Francis Engelmann, Suyu You, and Leonidas Guibas. Img2cad: Reverse engineering 3d cad models from images through vlm-assisted conditional factorization, 2025.
 - [31] Xumin Yu, Yongming Rao, Ziyi Wang, Zuyan Liu, Jiwen Lu, and Jie Zhou. Pointr: Diverse point cloud completion with geometry-aware transformers, 2021.
 - [32] Wentao Yuan, Tejas Khot, David Held, Christoph Mertz, and Martial Hebert. Pcn: Point completion network, 2019.
 - [33] Nannan Zhao and Shijie Chen. Industrial weld defect detection based on monocular depth estimation and dual-attention point cloud network. *Sensors (Basel)*, 26(11):3321, May 2026.
 - [34] Zuowei Zhu, Kévin Ferreira, Nabil Anwer, Luc Mathieu, Kai Guo, and Lihong Qiao. Convolutional neural network for geometric deviation prediction in

additive manufacturing. *Procedia CIRP*, 91:534–539, 2020. Enhancing design through the 4th Industrial Revolution Thinking.

A

Appendix 1

A.1 Deviation Prediction Results

Defect	Geometry MAE	Heatmap MAE	Baseline MAE
Bulge	0.96	0.7	1.312
Crack	0.8	0.573	0.924
Cut	0.96	0.778	1.032
Hole	1.02	0.749	1.145
Over-Extrusion	1.15	0.682	1.209
Under-Extrusion	1.75	1.226	1.643
Overall	1.02	0.74	1.16

Table A.1: Comparison of magnitude MAE (mm) per defect between baseline, geometry-only model and heatmap-enabled model at epoch 80

Defect	Geometry Model	Heatmap Model	Test points
Bulge	0.42	0.995	30900
Crack	0.8	1	13249
Cut	0.77	1	38505
Hole	0.8	1	16756
Over-Extrusion	0.96	1	24576
Under-Extrusion	0.74	1	5577
Overall	0.73	≥ 0.99	-

Table A.2: Comparison of Sign Accuracy on significant points between geometry-only model and heatmap-enabled model at epoch 80

A.2 Geometry Reconstruction Results

The following figures show one sample from each defect type, focusing on the defect region.

1. Crack Defect Sample
2. Cut Defect Sample
3. Hole Defect Sample
4. Over-Extrusion Defect Sample

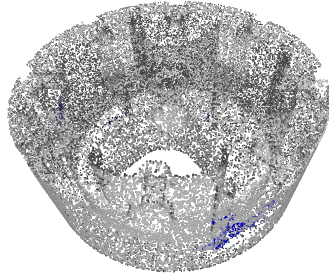


Figure A.1: Zoomed view of crack sample 009 for part 1M2

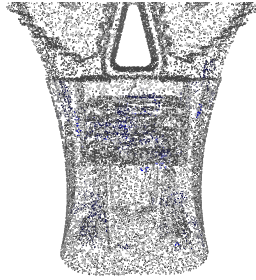


Figure A.2: Zoomed view of cut sample 011 for part 3M2C

5. Under-Extrusion Defect Sample

A.3 Quantification Results

Part	Defect	Area (mm ²)	Mean Depth (mm)	Volume (mm ³)
3M2C	Bulge	2642.1	0.82	2256.5
3M2C	Cut	4474.3	1.21	-5560
3M2C	Hole	2134.8	0.76	-1625.1
3M2C	Over-Extrusion	16315	1.24	19271.7
3M2C	Under-Extrusion	5549.2	1.55	-8678
1M2	Crack	2691.8	0.83	-2240.7
1M2	Hole	1449.7	1.01	-1501.5
1M3	Bulge	987.6	0.89	894.6
1M3	Crack	970.5	0.83	-832.6
1M3	Cut	1738.2	0.66	-1136.7
1M3	Hole	473.7	0.95	-475.5
4M2	Bulge	1307.1	1.19	1524.4
4M2	Cut	1953.2	0.67	-1319.8
4M2	Hole	1906.2	0.88	-1700.2
4M2	Over-Extrusion	6122.1	0.75	4649
4M2	Under-Extrusion	1187.8	1.59	-1986.9
2M2H	Bulge	2502.1	1.66	4466.1
2M2H	Cut	1827.4	0.82	-1581.5
2M2M	Bulge	2282.2	1.68	3878
2M2M	Hole	1430.7	1.14	-1707.6
3M1	Hole	616.1	1.06	-657.7

Table A.3: Three quantification measures by part and defect type

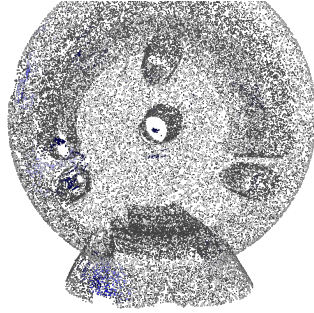


Figure A.3: Zoomed view of hole sample 004 for part 4M2

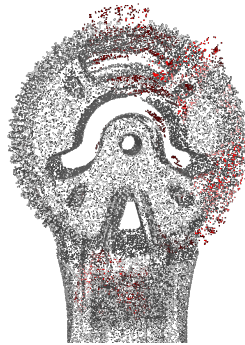


Figure A.4: Zoomed view of over-extrusion sample 014 for part 3M2C

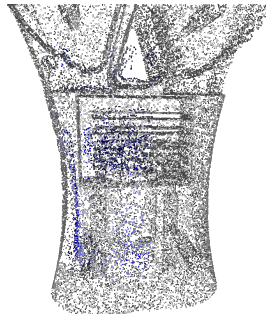


Figure A.5: Zoomed view of under-extrusion sample 001 for part 3M2C



CHALMERS
UNIVERSITY OF TECHNOLOGY