

Development of a Vision-based Monitoring System for Quality Assessment of 3D Printing

by

Jingdong Li

A thesis submitted in partial fulfilment for the requirements for the degree of
MSc (by Research) at the University of Central Lancashire

May 2023

RESEARCH STUDENT DECLARATION FORM

Type of Award _____ MSc (by Research) _____

School _____ University of Central Lancashire _____

Sections marked * delete as appropriate

1. Concurrent registration for two or more academic awards

Either *I declare that while registered as a candidate for the research degree, I have not been a registered candidate or enrolled student for another award of the University or other academic or professional institution

or ~~*I declare that while registered for the research degree, I was with the University's specific permission, a *registered candidate/*enrolled student for the following award:~~

2. Material submitted for another award

Either *I declare that no material contained in the thesis has been used in any other submission for an academic award and is solely my own work

or ~~*I declare that the following material contained in the thesis formed part of a submission for the award of:~~

(state award and awarding body and list the material below):

3. Collaboration

Where a candidate's research programme is part of a collaborative project, the thesis must indicate in addition clearly the candidate's individual contribution and the extent of the collaboration. Please state below:

4. Use of a Proof-reader

Either ~~*The following third-party proof-reading service was used for this thesis _____ in accordance with the Policy on Proof-reading for Research Degree Programmes and the Research Element of Professional Doctorate Programmes.~~

~~_____ A copy of the confirmatory statement of acceptance from that service has been lodged with the Academic Registry.~~

or *No proof-reading service was used in the compilation of this thesis.

Signature of Candidate

Jingdong Li

Print name: Jingdong Li

Development of a Vision-based Monitoring System for Quality Assessment

Acknowledgements

I am grateful for the opportunity to write this thesis, and I am grateful to everyone who has supported and helped me along the way. I would like to express my sincerest appreciation to my research director, Dr. Wei Quan, who has been a constant source of inspiration and guidance. Their support, encouragement, and expertise have been invaluable in shaping my research and helping me to bring this thesis to fruition. I am also deeply grateful to my advisor, Dr. Hadley Brooks and Prof. Lik-Kwan Shark, for their insightful comments and suggestions that have greatly improved this work.

I would like to thank the University of Central Lancashire for providing me with the resources and opportunities to pursue my research interests. I am also grateful to my colleagues and friends who have provided me with support and encouragement throughout this journey.

Lastly, I would like to express my appreciation to my family, especially my parents, for their unwavering love and support. This thesis would not have been possible without their encouragement and sacrifices. I am grateful to everyone who has helped me along the way.

Abstract

Additive Manufacturing (AM), also known as 3D printing, is a process of manufacturing parts and components by adding successive layers of material on top of each other until the final shape is achieved. The research target of this project is Fused Filament Fabrication (FFF), which is a specific type of Additive Manufacturing technology. FFF uses a filament of thermoplastic material, which is melted and extruded, then deposited layer by layer to create a 3D object. However, FFF has some limitations that need to be considered. For instance, the printing process can be time-consuming, and errors such as misalignment and incorrect slicing can occur, leading to complete failure and wasted time and material.

This thesis presents a vision-based monitoring system for FFF 3D printing quality assessment. The proposed system includes a simulation tool that generates simulated images of printed layers, along with feature extraction methods for assessing the size, shape and infill density of printed objects. The proposed system utilizes background subtraction for isolating the printed object from the background and estimating its size through pixel length analysis and bounding box calculation. The shape analysis of the printed objects is performed using the Fourier-Mellin transform (FMT) method. Moreover, the infill density is computed by combining foreground extraction and image thresholding methods, utilizing both camera and simulated images. The proposed system is able to analyse and examine the quality of 3D printing during the printing process and identify the defective printed object when deviates of 5 percent is detected in terms of the size, shape, and density of the printed object, alert the user to terminate the entire process and save time and cost. This new monitoring system provides an effective solution to improve the quality and efficiency of FFF 3D printing.

Table of Contents

Acknowledgements.....	3
Abstract.....	4
Table of Contents	5
List of Figures	7
List of Tables	9
List of Abbreviations.....	10
1. Introduction	11
1.1 Background.....	11
1.2 Aim of Research.....	15
1.3 Research Contributions.....	17
1.4 Thesis Organisation	19
2. Overview of 3D Printing and Monitoring System	20
2.1 Challenge of 3D printing.....	20
2.1.1 Object not sticking	20
2.1.2 Quality of surface.....	23
2.1.3 Clogged Nozzle	25
2.1.4 Warping Deformation.....	26
2.2 Overview of monitoring system	28
2.2.1 Non-vision-based system	29
2.2.2 Vision-based system	32
2.2.3 Summary of monitor systems	35
3. System Design.....	37
3.1 Camera parameter estimation.....	38
3.2 Print Simulation	42
3.3 Feature extraction	50
4. Feature Evaluation.....	65
4.1 Evaluation of size	65
4.2 Evaluation of infill density.....	67

4.3 Evaluation of shape.....	68
5. Experimental Analysis.....	69
5.1 Test on different shape	69
5.2 Test on different layer	72
5.3 Entire test.....	75
5.4 Summary.....	78
6. Conclusions and Future Work.....	80
References	82
Appendix.....	87

List of Figures

Figure 1.1: Schematic diagram of the FFF 3D printing process	13
Figure 1.2: Type of error in 3D printing.....	14
Figure 1.3: The proposed vision-based monitoring system for quality assessment .	16
Figure 2.1: A failed 3d print of spaghetti.....	22
Figure 2.2: Object displacement during printing	23
Figure 2.3: Quality of surface	25
Figure 2.4: Extruder of 3D printer	26
Figure 2.5: Warping of object.	27
Figure 2.6: Schematic of warping deformation	28
Figure 2.7 Schematic representation of the X-ray CT sample and its print patterns.	30
Figure 2.8 3D rendering of the voids in the carbon fibre reinforced composite.	31
Figure 2.9 Profiles of the different steps of the measured sample	32
Figure 2.10 Multi-camera system experimental setup	34
Figure 3.1: Flow diagram of proposed vision-based monitoring system.....	38
Figure 3.2: Illustration of the experimental printer and camera setup.....	40
Figure 3.3: Mean reprojection error per Image.....	41
Figure 3.4: Position relative of camera and print platform	42
Figure 3.5: Overview of three.js system	48
Figure 3.6: Simulated Image with Material and Lighting for Printed Object.	49
Figure 3.7: Simulated object.....	50
Figure 3.8: Flowchart of foreground extraction.....	51
Figure 3.9: Relationship of threshold value and infill density.....	54
Figure 3.10: Result of thresholding.....	54
Figure 3.11 Sequential processing in Fourier-Mellin Transform	56
Figure 3.12 An example of translation invariant in Fourier Transform	57
Figure 3.13 An example of rectangular coordinates to polar coordinates conversion	59
Figure 3.14 An example of rotation effect.....	59
Figure 3.15 An example of scale effect	60
Figure 3.16 An example of translation invariant of Fourier Transform	62
Figure 3.17 An illustration of Fourier-Mellin transform.....	64

Figure 4.1 Size Instruct of camera image and simulated image through bounding box	66
Figure 4.2 An example of the density calculation	68
Figure 4.3 FMT images of the camera and simulate image	68
Figure 5.1: Test objects OA (upper-row) and OB (lower-row)	73
Figure 5.2: First layer size extraction.....	76
Figure 5.3: Middle layer infill density extraction	76
Figure 5.4: Last layer shape extraction	77
Figure 5.4: Plot of the feature evaluation parameters as the number of printed layers.	78

List of Tables

Table 3.1: Most common G-Codes/M-Codes used in 3D printing.	45
Table 5.1: Summary of inherent displacements of 3D printing	71
Table 5.2: Print quality assessment for test object OA.	74
Table 5.3: Print quality assessment for test object OB	75

List of Abbreviations

ABS	Acrylonitrile Butadiene Styrene
AM	Additive Manufacturing
AR	Augmented Reality
CAD	Computer-aided design
FFF	Fused Filament Fabrication
FMT	Fourier-Mellin Transform
HTML	Hypertext Markup Language
JS	JavaScript
LCD	Liquid Crystal Display
PETG	Polyethylene Terephthalate Glycol
PLA	Polylactic Acid
RANSAC	Random Sample Consensus
RST	Rotation, scaling and translation
SIFT	Scale-Invariant Feature Transform
SLA	Stereolithography
SLS	Selective Laser Sintering
STL	Stereolithography

1. Introduction

1.1 Background

Additive Manufacturing (AM), also known as 3D printing, is a process of manufacturing parts and components by adding successive layers of material on top of each other until the final shape is achieved (Dancel, 2019). AM has a wide range of applications, from creating prototypes and tooling in the manufacturing industry to producing custom medical implants and prosthetics, jewellery, and even food (Haleem & Javaid, 2019). However, the technology is still evolving, and there are challenges to be addressed, such as the cost of materials, the speed of the printing process, and the quality and consistency of the finished products (Fang & Kumar, 2019).

There are various types of 3D printing technologies available for creating objects using additive manufacturing (AM). These include Fused Filament Fabrication (FFF), which involves melting and extruding a thermoplastic filament layer by layer; Stereolithography (SLA), which solidifies liquid resin using a laser layer by layer; Selective Laser Sintering (SLS), which utilizes a laser to sinter powdered materials layer by layer, resulting in objects made of materials such as plastics or metals; and Liquid Crystal Display (LCD) printing technologies, which involve projecting an image or pattern onto a photosensitive resin using an LCD screen, which is then cured or hardened by a light source such as UV light (Shahrubudin & Ramlan, 2019).

Among these technologies, FFF stands out as a popular method within additive manufacturing. One primary advantage lies in its affordability. In comparison to other additive manufacturing technologies, FFF 3D printers generally possess a lower cost, thus broadening accessibility to a diverse spectrum of users ranging from hobbyists to professionals. Another benefit is the expansive array of thermoplastic materials that FFF accommodates, encompassing ABS, PLA, nylon, polycarbonate, and more (Dey et al., 2021). This adaptability facilitates the incorporation of distinct material properties, an assortment of colours, and diverse applications. Furthermore, FFF's user-friendliness underscores its appeal. The operational parameters of FFF printers necessitate only a minimal configuration, thereby enabling designers and engineers to engage in rapid prototyping. Nonetheless, FFF does exhibit certain constraints. The

inherent layer-by-layer methodology of FFF may yield diminished resolution in comparison to alternative 3D printing techniques, rendering it unsuitable for intricate or intricate models. Additionally, a trade-off emerges between print speed and quality; striving for heightened printing speeds can compromise the quality of the output, while prioritizing superior quality may extend production timelines. Thus, a judicious equilibrium between swiftness and excellence becomes imperative. Depending on the intended final outcome and mechanical attributes, post-processing steps such as sanding, painting, or coating might be requisite to achieve the intended outcome for FFF prints (Bochmann et al., 2015).

The FFF process consists of three phases: CAD software design, slicing software conversion, and the 3D printing process (Hu, 2017). As shown in Figure 1.1, to start, a 3D model is created using CAD software and then imported into slicing software. The slicing software converts the 3D model into machine-readable code for the 3D printer. This code is a set of precise commands that guide the 3D printer's movements and operations, including the extrusion of material, nozzle movement, and platform movement. It is a digital representation of the object's geometry and how it should be constructed layer by layer. The Polylactic Acid (PLA) thermoplastic material used in this project is heated and extruded from the nozzle, solidifying on the build platform to form the first layer of the object. The rest of the layers are built upon the previous one through horizontal movement of the nozzle and vertical movement of the platform. In situations where the design has overhanging features, support material is used to prop up these areas, enabling the successful printing of complex objects (Medelli et al., 2009).

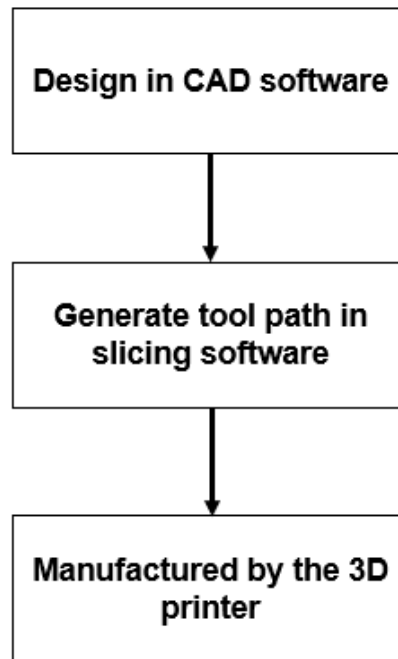


Figure 1.1: Schematic diagram of the FFF 3D printing process

Despite the fast-paced advancement of FFF techniques, several challenges persist, including filament quality issues, first layer adhesion problems, partial warping, wire drawing, and incorrect printing parameters (Bochmann et al., 2015). Real-time quality control of printed objects is a technical challenge in additive manufacturing. The main sources of errors in FFF are material-related and process-related errors, and some classic error types are depicted in Figure 1.2.



Figure 1.2: Type of error in 3D printing (Kerry, 2018)

The errors in the FFF system include surface quality, dimensional accuracy, and material adhesion. Surface quality is impacted by various factors such as the speed of print moves, bed temperature, extruder temperature, and the materials used in the printing device's construction and its accuracy (Fastowicz et al., 2019). Maintaining dimensional accuracy of printed objects is another challenge, even with consistent parameter settings and printing environment. During the extrusion process, the filament cools rapidly from the glass transition temperature to the cavity temperature, which can cause internal stress and deformations, potentially leading to failure of the print (Zeltmann et al., 2016). Additionally, poor material adhesion can negatively impact mechanical performance. During the printing process, inadequate interlayer adhesion can result in geometric deformations of printed objects.

The 3D printing process can be lengthy, meaning it takes a lot of time to complete. During this time, the 3D printer will produce noticeable noise, which may be disruptive in quiet or shared workspaces. Because of this noise, individuals or organizations often choose to locate 3D printers in separate areas or dedicated spaces to minimize disruption to nearby workers. This separation can pose challenges for physical oversight or monitoring of the 3D printing process. Although some retailers offer remote monitoring options, the extended printing time can still make it challenging for users to keep track of the process. Without supervision, errors can lead to significant time and material waste, depending on how quickly the user discovers the problem (Lyngby et al., 2017). To address this issue and improve quality control while reducing costs, various monitoring systems have been developed, including both vision-based and non-vision-based methods. This project specifically chose to develop a vision-based system, as such methods can quickly and efficiently detect printing defects.

1.2 Aim of Research

The aim of this project is to develop a 3D printing monitoring system that assesses the quality of printed objects. As shown in Figure 1.3, the proposed system captures images of the object from above the printer using a PC-controlled camera, taking pictures at the end of each layer of the print. Additionally, simulation software generates simulated images of each layer for comparison. The system evaluates the printed images by simulating them to identify errors, ultimately saving time and materials by alerting the user to printing errors and aborting the printing process.

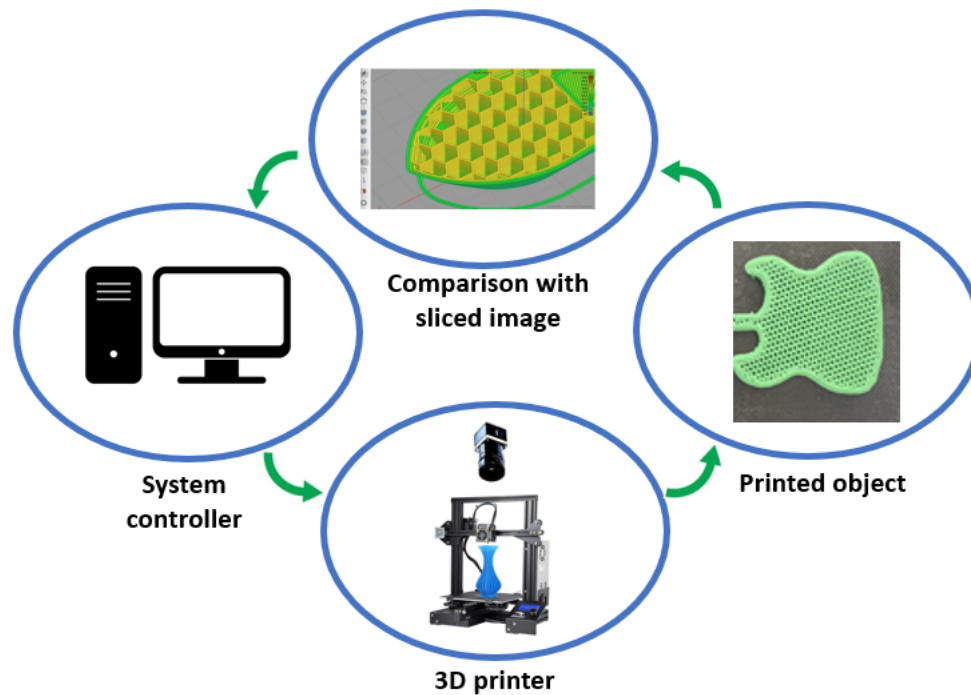


Figure 1.3: The proposed vision-based monitoring system for quality assessment

The system proposed is expected to meet the following requirements:

- Developed a simulation tool that can generate simulated images of printed layers.
- Extract size, shape and infill pattern features from simulated and camera images of printed objects to evaluate the quality of the printing process.
- Compare the extracted features to evaluate the quality of printed objects.
- When any of the detection errors related to size, shape, or infill density exceeds the 5% threshold, it is determined that the part is defective. Subsequently, an output is provided for the user to decide whether to terminate the process, thereby saving the user time and cost.

1.3 Research Contributions

The original contributions of the research work presented in this thesis can be summarised as follows:

- Development of simulation tools: A simulation tool is developed that uses the G-code generated by slicing software and the camera position obtained through camera parameter estimation to generate simulated images of each layer of the printed object.

- Evaluation of size: The size of a printed object can be determined using two processing stages. Firstly, background subtraction is used to isolate the object from the background. Secondly, the size of the printed object can be estimated by analysing the length represented by each pixel and examining the bounding box obtained through blob analysis.

- Evaluation of shape: The shape of printed objects is extracted from camera images by using background subtraction methods, and the shape feature of printed objects is evaluated by using Fourier-Mellin Transform (FMT) method.

- Evaluation of infill density: The infill density of printed objects is calculated from camera images and simulated images using foreground extraction and image thresholding method. The evaluation of infill density is based on the calculation of the number of pixels that represent the infill material.

To simulate the process of 3D printing, the camera calibration method is used to determine the position of the camera in the scene. The position of the camera allows the simulation image to maintain the same perspective as camera image.

In this project, size, shape and infill density were chosen as the focus of the system due to the correlation of these factors with print quality. Size, shape and infill density are key factors that directly affect the quality and functionality of a 3D printed object. Monitoring these parameters can assess whether a printed object meets design

specifications and standards. Also, size variance, shape irregularities, and infill density issues are common pitfalls in 3D printing. Detecting and evaluating these aspects can help identify potential printing problems early in the process. In addition, attention was paid to the choice of size, shape, and fill density to match the interests and concerns of the user. These factors are often critical for users who wish to 3D print an exact match to the intended design. Most importantly, size and shape can be assessed relatively directly using image analysis techniques, while infill density is critical to assessing the structural integrity of printed objects.

Regarding why layer thickness is not chosen as a primary focus, it is important to recognize that layer thickness is undoubtedly a critical parameter in 3D printing. However, accurately gauging layer thickness solely through image analysis methods can be inherently challenging due to the limitations of such techniques. Accurate layer thickness measurement often necessitates direct physical measurements of the printed layers, which may require specialized equipment or measurement approaches beyond the scope of the system described in this thesis.

1.4 Thesis Organisation

Chapter 1: An introduction to the entire project is provided, including the research motivation, problem description, a preview of the proposed method and aim of the research.

Chapter 2: A literature review is given that describes the problem in additive manufacturing and an overview of the monitoring system.

Chapter 3: This chapter focuses on explaining the development process of the monitoring system. Additionally, it discusses various methods to extract image features, which are essential for the successful functioning of the monitoring system.

Chapter 4: This chapter provides insights into the performance and outcomes of the monitoring system including the result of research and experimental analysis.

Chapter 5: A summary of the research is stated and is followed by outlining the potential improvement for future work.

2. Overview of 3D Printing and Monitoring System

In this overview, the common errors that can occur during the printing process will be explored. With the aim of improving quality control and reducing waste, the importance of process monitoring will also be explored, including a discussion of the different surveillance systems available, such as vision-based and non-vision-based approaches. The advantages and limitations of each will be examined, culminating in a discussion of why a vision-based approach was chosen for this project.

2.1 Challenge of 3D printing

To improve the quality and minimize waste in 3D printing, it's important to understand and address common errors that can occur during the process. These errors include poor filament quality, first layer adhesion issues, warping, stringing, incorrect printing parameters, and more (Anitha & Radhakrishnan, 2001). To prevent these errors, it's crucial to control input process parameters, select appropriate materials and structures, and perform regular maintenance on 3D printers. Some of the most common errors that can occur during 3D printing include poor first layer adhesion, over/under extrusion, layer offset, clogged extruder, warpage, and spots and pimples (Oropallo & Piegl, 2016). By identifying and addressing these errors, and implementing proper monitoring and control measures, the quality and consistency of 3D printed objects can be improved, leading to increased practical use and reduced waste.

2.1.1 Object not sticking

One common cause of 3D printing failure is when the object does not properly adhere to the build plate during the printing process. This issue, known as object not sticking, can lead to warping, distortion, or even detachment of the printed object from the build plate. There are several factors that can contribute to this problem:

- **Incorrect bed levelling:** If the build plate is not level, the distance between the nozzle and the plate might be inconsistent, leading to uneven deposition of the filament and poor adhesion.

- Inadequate bed temperature: Many 3D printers have heated build plates to promote better adhesion. If the bed temperature is too low, the filament may not adhere properly to the surface.
- Dirty or oily build surface: Contamination of the build plate, such as dust, dirt, or oils from human skin, can reduce the adhesion of the printed material. It is essential to clean the build plate regularly to ensure good adhesion.
- Insufficient "squish": When the first layer is deposited, a slight "squish" or compression of the filament against the build plate is necessary for good adhesion. If the nozzle is too far from the build plate, the filament will not be compressed enough, leading to poor adhesion.
- Incorrect first layer settings: The settings for the first layer, such as print speed, layer height, and extrusion rate, can impact adhesion. Slower print speeds and lower layer heights for the first layer can improve adhesion.
- Material properties: The material's characteristics can indeed influence its adhesion to the build platform. For instance, securing adequate bed adhesion with ABS can prove to be a formidable task, often necessitating the utilization of heated beds and supplementary adhesion aids like adhesive glue or tape. In contrast, PLA exhibits lower heat resistance in comparison to materials like ABS. When subjected to elevated temperatures, PLA undergoes softening and deformation, which can compromise its adhesion properties. Additionally, owing to the flexibility of PU, ensuring reliable bed adhesion can be a nuanced endeavour, placing significant emphasis on meticulous bed levelling and thorough surface preparation.

When a printed object fails to adhere to the build platform, the "spaghetti" phenomenon may occur. As depicted in Figure 2.1, the unattached printing material accumulates next to the printed object, resembling spaghetti. This situation can potentially damage the printer without anyone being aware of it. To address this issue, Baumann et al. (2016) proposed a solution that detects errors by identifying the offset of objects.

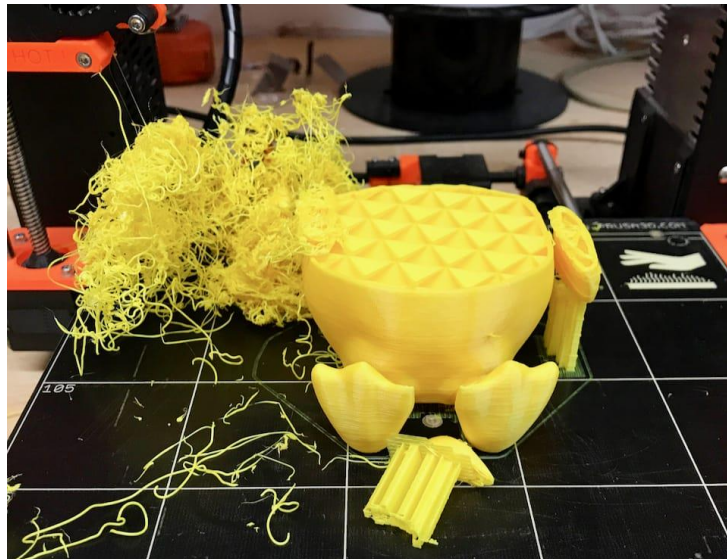


Figure 2.1: A failed 3d print of spaghetti. ALL3DP (2019)

The proposed solution holds the potential to significantly enhance the accuracy of 3D printing and minimize the number of failed prints by detecting displacement errors in real time. This leads to time and cost savings for users and improved efficiency in the 3D printing process. Additionally, the utilization of computer vision algorithms and image processing techniques makes the solution both accessible and cost-effective, as it does not necessitate the installation of extra sensors or modifications to the printer firmware. As shown in Figure 2.2, which shows the deviations that occur in printed objects, the proposed solution uses computer vision algorithms for displacement error detection. The method involves real-time detection of displacement errors, which can significantly improve the accuracy of 3D printing and reduce the occurrence of printing failures. This not only saves users a lot of time and money, but also improves the overall efficiency of the 3D printing process. Furthermore, the effectiveness of this strategy is highlighted by its reliance on computer vision algorithms and image processing techniques. These technologies make the solution easy to use and cost-effective, as they do not require additional sensors or modifications to the printer's firmware. provides a promising strategy for online quality control in 3D printing.

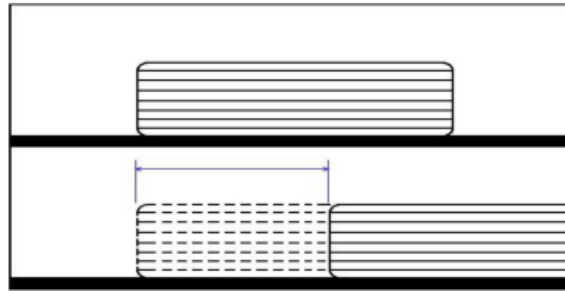


Figure 2.2: Object displacement during printing (Baumann, 2016).

2.1.2 Quality of surface

Another cause of 3D printing failure is the quality of the surface of the printed object. Poor surface quality can manifest in various ways, such as roughness, inconsistent layers, or visible seams. Several factors can contribute to this issue:

- **Inadequate layer resolution:** The layer height setting in the slicing software determines the thickness of each layer in the printed object. If the layer height is too large, the individual layers may become more visible, resulting in a rough or uneven surface.
- **Inconsistent extrusion:** Variations in the extrusion rate, either due to inconsistencies in the filament diameter or a clogged extruder nozzle, can cause irregularities in the surface of the printed object. Properly maintaining the extruder and using high-quality filament can help mitigate this issue.
- **Print speed:** Printing at excessively high speeds can compromise the surface quality of the printed object, as it may not allow the material to properly bond to the previous layer. Slowing down the print speed can improve the surface finish.
- **Inadequate cooling:** Insufficient cooling during the printing process can cause the printed layers to warp or sag, affecting the surface quality. Ensuring proper cooling, either by using a cooling fan or adjusting the printing temperature, can help maintain a smooth surface.
- **Seams and Z-scar:** When the extruder completes a layer and moves to start the next one, it can leave a visible seam, or a small vertical line known as a Z-scar on

the surface of the object. Adjusting the slicer settings, such as retraction distance and coasting, can help minimize the visibility of seams and Z-scars.

- **Nozzle size:** The size of the nozzle determines the minimum achievable feature size and layer thickness. A smaller nozzle allows for finer details and higher resolution, but it also increases printing time. On the other hand, a larger nozzle can speed up the printing process but may sacrifice fine details.

The poor surface quality of 3D-printed objects can have multiple negative effects on the overall usability, appearance, and functionality of the final product. These may include reduced structural integrity, impaired functionality, and the need for additional post-processing steps. Figure 2.3 shows the surface quality of printed objects in different situations. To assess the quality of 3D printed object surfaces, Fastowicz et al. (2019) proposed a solution based on the entropy of depth maps. A depth map is a 2D representation of the 3D surface of a printed object. It encodes information about the distance or depth of each point on the object's surface relative to a reference point. Entropy is a concept in information theory that measures the degree of disorder or randomness in a data set. In this case, the dataset is a depth map of the surface of a 3D printed object. The proposed solution evaluates the display quality by converting a picture of the surface of a printed object into a depth map and calculating the entropy.

Furthermore, the proposed solution can be integrated into the control software of a 3D printer to provide real-time feedback and control over the printing process. It can be fine-tuned to improve accuracy and adapted to different materials. This system also serves as the foundation for further research in developing more advanced and automated 3D printing quality control techniques. By providing an excellent basis for automated visual assessment and improvement of 3D printed surface quality, this method helps reduce the time and costs associated with manual quality control, resulting in more efficient and accurate 3D printing.

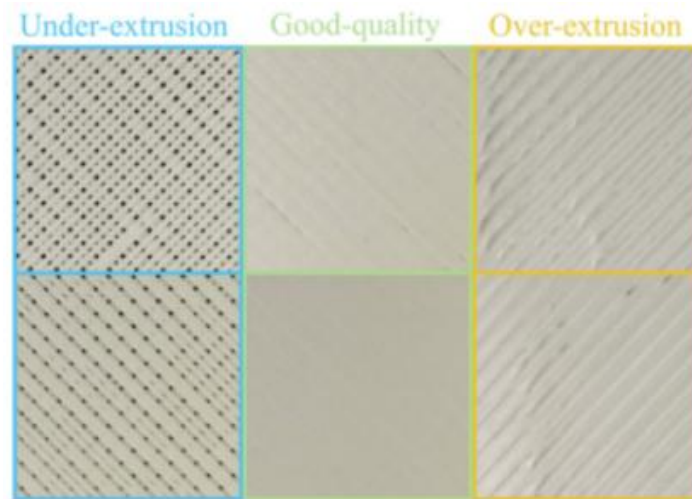


Figure 2.3: Quality of surface (Fastowicz et al., 2019)

2.1.3 Clogged Nozzle

A clogged nozzle is another common cause of 3D printing failure. During the printing process, the thermoplastic filament is melted and extruded through the printer's nozzle. Over time, small particles, dust, or even residual filament material can accumulate inside the nozzle, causing it to become partially or fully blocked. When the nozzle is clogged, the flow of molten filament is restricted or entirely obstructed, leading to incomplete or failed prints.

As depicted in Figure 2.4, the extruder is a complex assembly composed of several components, including the motor, gears, and idlers. The motor provides the force needed to advance the filament, while the gears ensure a consistent feed rate. The idler maintains pressure on the filament as it is fed into the hot end. The hot end melts the filament and deposits it layer by layer onto the print bed, forming the final printed object.

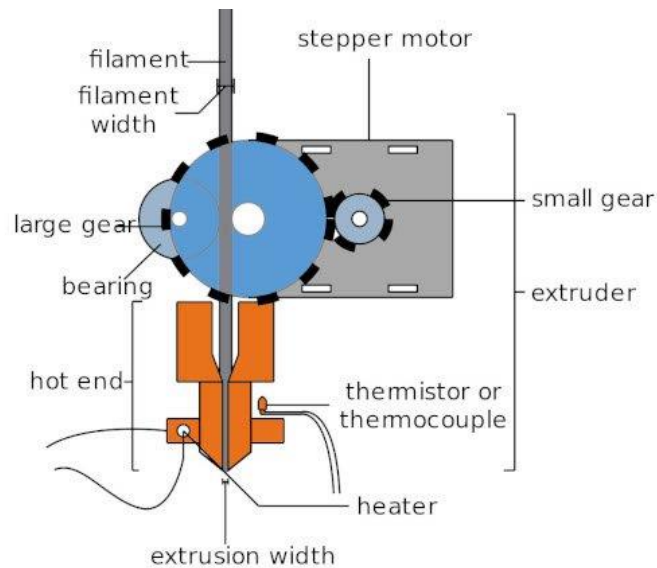


Figure 2.4: Extruder of 3D printer (Nikhil, 2023)

A clogged nozzle in 3D printing leads to insufficient filament extrusion, resulting in incomplete prints or missing layers in the printed object. This issue can render the end product unusable, necessitating reprinting and wasting time and materials. In more severe cases, it can cause significant damage to the extruder components. If the flow of material is interrupted in the heated part of the extruder where molten material is present, the motor continues to extrude material through the system. This causes the hot end to become overfilled with molten material, which eventually passes through the thermal break and moves up into the heat sink, where it cools down. To address this problem, Nuchitprasitchai et al. (2017) have proposed using optical sensors to detect nozzle clogging. By monitoring the performance of the hot end and nozzle, the 3D printing process can be optimized to enhance print quality and reduce the likelihood of print failures.

2.1.4 Warping Deformation

Warping deformation is a common cause of 3D printing failure. As shown in Figure 2.5, it occurs when the printed object's edges or corners lift and curl away from the build plate during the printing process, resulting in a distorted and unusable final product. This issue is primarily caused by the uneven cooling of the printed material, leading to internal stresses within the object.

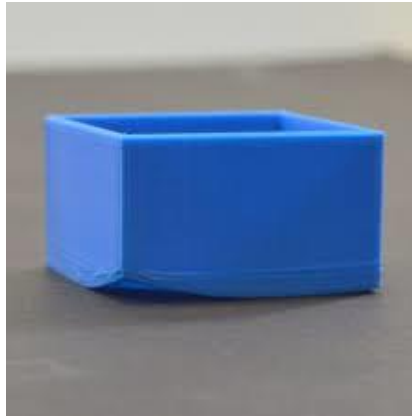


Figure 2.5: Warping of object.

As shown in Figure 2.6, warping usually occurs from both sides of the object. Warping can be caused by several factors such as the material used, the temperature of the print bed, the speed at which the material is deposited, and the ambient temperature of the printing environment. To minimize warping, the use of a heated print bed, adhesion materials like a blue painter's tape, and adjusting the printing temperature and speed can help. These features can be achieved by using cooling fans, slowing down the cooling process by reducing the print speed, or using materials with lower shrinkage. An example of using a material with a lower shrinkage rate to minimize warping in 3D printing is Polyethylene Terephthalate Glycol (PETG). PETG is known for its lower shrinkage compared to some other common 3D printing materials like ABS. The use of a good adhesion material can prevent issues like print detachment and warping. An example of such a material is blue painter's tape or Kapton tape. These types of tape can be applied to the print bed before printing. They provide a surface that the printed object can adhere to effectively, reducing the likelihood of it coming loose during the printing process. This is especially useful when working with materials like ABS, which can be prone to warping without proper adhesion. It's worth noting that while a cooling fan can help reduce the time it takes for the filament to cool, it doesn't necessarily mean that faster cooling is better. The speed at which the cooling fans operate can affect print quality, especially with certain materials. Some 3D printing materials benefit from controlled and gradual cooling to prevent issues such as layer separation, curling or warping. For example, when printing with PLA, little or no active cooling is usually required in the first few layers to ensure good adhesion

between the layers. However, once the print goes beyond these initial layers, the cooling fans can be turned on to help with overhangs and finer details. In contrast, materials such as ABS may require more cooling to prevent warping, but excessive cooling can introduce issues such as layer adhesion issues and surface defects. Therefore, the ideal cooling fan speed depends on the specific material used and the specific requirements of the print. It's a balance between cooling to prevent problems and proper conditions to maintain proper layer adhesion and print quality. Additionally, monitoring the temperature of the print bed and ensuring it stays within the recommended range can help prevent warping. It's important to note that different materials and printing environments may require different strategies to minimize warping, so understanding the underlying causes and experimenting with different solutions is key to preventing this issue.

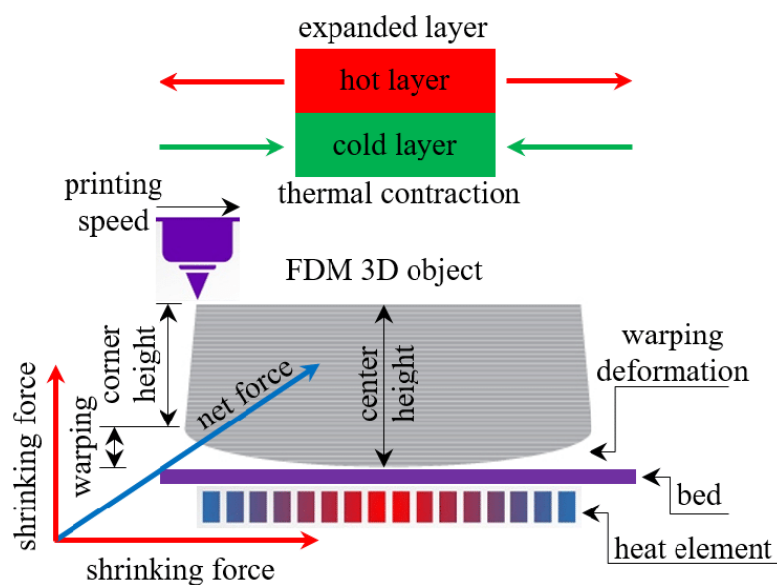


Figure 2.6: Schematic of warping deformation (Alsoufi, 2017)

2.2 Overview of monitoring system

There are two main categories of 3D printing monitoring systems: vision-based and non-vision-based methods. Non-vision systems use technology which is frequently used in non-destructive test systems, but may not capture all errors, particularly those related to the appearance of the printed object. Vision-based systems, on the other

Development of a Vision-based Monitoring System for Quality Assessment

hand, utilize cameras to record the printing process, allowing for the identification of errors related to object appearance, including warping, poor adhesion, and layer shifting. Vision-based methods can also offer valuable insights into the quality of the final printed object (Cheng & Jafari, 2008). While both approaches have their own strengths and limitations, a detailed description of each will be provided below.

2.2.1 Non-vision-based system

In this section, various non-vision-based surveillance systems are studied, most of which employ non-destructive methods to detect defects, such as Soete et al. (2019) employed X-ray tomography and ultrasound imaging to detect embedded defects. Busch et al. (2014) used an electromagnetic method called terahertz non-destructive testing to detect deformation inside objects. Fase et al. (2016) proposed a method of scanning by laser, which uses the laser as a sensor to measure the size of the object in the process. It detects errors and provides corrections at an early stage in the process. Sitthi-Amorn et al. (2015) Internal defect detection of 3D printed objects by applying X-ray tomography and ultrasonic imaging to detect embedded defects and change printing orientation. Ceruti et al. (2017) used augmented reality during the quality monitoring process to superimpose a virtual model on the printed object, allowing the identification of shape differences between the model and the printed object at different printing stages. The details of non-vision-based monitor system will be described below.

X-ray CT works by capturing X-ray images from different angles and using computer algorithms to reconstruct the 3D structure of an object from those images. The system developed by Soete et al. (2019) uses X-ray CT to detect internal imperfections in 3D printed objects, including voids, porosity, and other structural defects. The advantage of this approach is that it is non-destructive, allowing the assessment of the quality of the 3D printed object without damaging it. The results of this study showed that X-ray CT can effectively detect internal imperfections in 3D printed objects and could be a useful tool for quality control in the 3D printing industry.

The X-ray CT system uses X-rays to produce detailed images of the internal structure of an object, which is used to detect the internal imperfections in 3D printed objects.

Development of a Vision-based Monitoring System for Quality Assessment

As shown in Figure 2.7, The X-ray CT system rotates the sample 360° and captures images from different angles, which are then reconstructed into a 3D model. This system is beneficial for identifying the internal defects in 3D printed objects that cannot be seen with the naked eye. By using this method, it is possible to perform a comprehensive evaluation of the quality of 3D printed objects and to make improvements to the printing process if necessary.

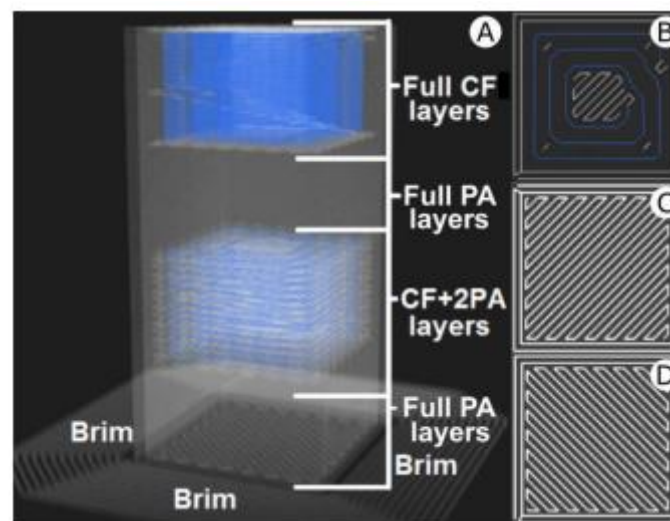


Figure 2.7 Schematic representation of the X-ray CT sample and its print patterns.(Soete et al., 2019)

The results showed that the system was able to accurately detect internal defects and imperfections in the 3D printed objects, including voids and material thickness variations. The proposed method can be used as a tool for quality control and defect detection in 3D printing processes and can provide valuable information for improving the print quality and reliability of the objects.

This system helps in assessing the internal quality of the 3D printed object. The results shown in Figure 2.8 indicate that the X-ray CT can detect the presence of voids, inclusions and layer separation within the 3D printed objects. This method offers a non-destructive way to monitor the quality of the internal structure of the printed objects and can be used as a tool for quality control in 3D printing.

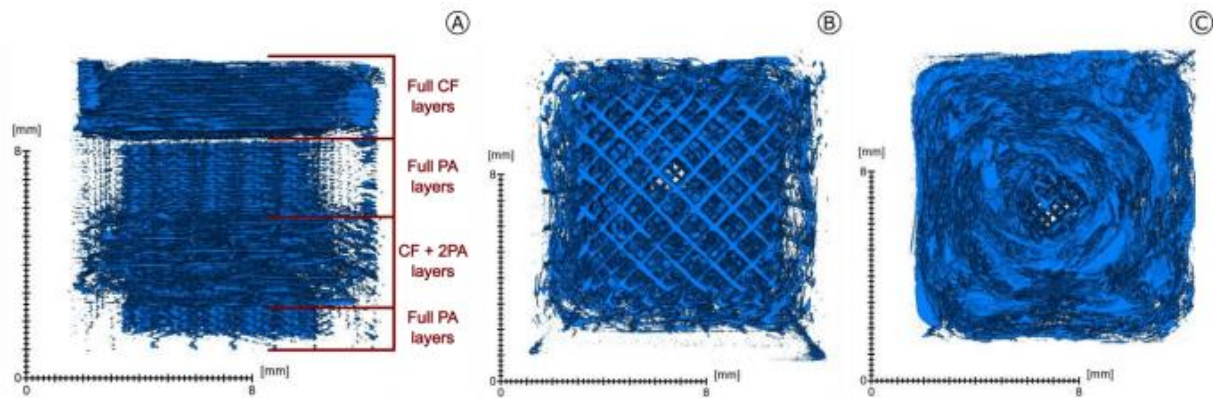


Figure 2.8 3D rendering of the voids in the carbon fibre reinforced composite. (Soete et al., 2019)

X-ray CT imaging provides detailed information about the internal structure of 3D printed objects, which can be used to evaluate its quality. The system uses thresholding and image analysis techniques to extract features from the CT scans and assess the porosity, density, angular variation, orientation of the fill pattern, and the presence of voids in the printed object.

The proposed system in Fase et al. (2016) uses laser scanning to monitor the printing process by measuring the size of objects during the printing. The laser scans the object at a high frequency and produces a point cloud data that represents the object's shape. The system then compares the scanned point cloud data with a reference model to detect deviations and errors in the printing process. If deviations are detected, the system provides correction in real-time to minimize the error and improve the quality of the printed object. This system provides an effective and non-invasive method for monitoring the 3D printing process and ensuring the quality of the final product.

This laser scanning system is capable of measuring the size of 3D printed objects during the process. The algorithm consists of several steps to detect the laser line, find the platform, detect the edge of the track, and calculate the height. The accuracy of the measuring system has been verified, with a maximum deviation of 100 μ m. The data obtained from the laser scanning is used to develop a triangulation system for error detection and correction in an early stage of the printing process.

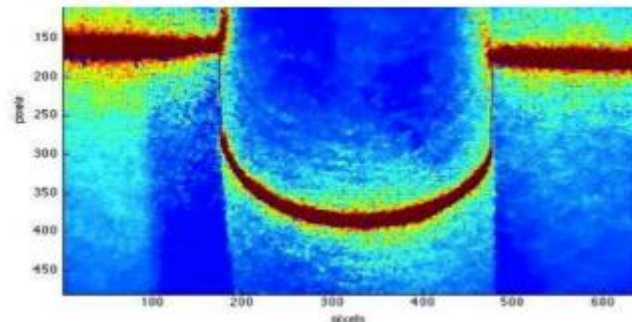


Figure 2.9 Profiles of the different steps of the measured sample(Fase et al., 2016)

2.2.2 Vision-based system

In this section, different vision-based monitoring systems for 3D printing are explored. These systems can be categorized into single-camera systems and multiple-camera systems based on the number of cameras used for monitoring.

For single-camera systems, Lyngby et al. (2017) proposed a real-time method for detecting printing failures using a single camera setup. The system utilizes an oblique placement of the camera over the printed object, with light sources positioned behind the camera to provide appropriate lighting conditions. To aid in image segmentation, captured images undergo a colour space conversion from RGB (red, green, blue) to HSV (hue, saturation, value). The HSV colour model separates the hue component, which represents primary colour information, from the saturation and value components, which represent colour intensity and lightness. By leveraging the hue component, the system achieves robust and efficient object differentiation based on colour attributes. Moreover, the HSV colour model demonstrates greater resilience to changes in lighting conditions compared to the RGB colour model. This resilience arises from the stability of the hue component under different lighting conditions, while the luminance component compensates for variations in brightness. Consequently, segmentation based on HSV values ensures reliable and consistent performance across diverse lighting environments. To generate simulated images, a perspective projection and a pinhole camera model are utilized to obtain the rotation matrix. This matrix facilitates the transformation from image point to world coordinate point, establishing a world coordinate system based on the printing platform. Accordingly, the simulated image can be rendered in accordance with this defined coordinate

Development of a Vision-based Monitoring System for Quality Assessment

system. Ultimately, by comparing shape feature differences between captured images and rendered images, the system can detect incomplete prints, layer shifts, or errors in material deposition.

Similarly, Nuchitprasitchai et al. (2017) proposed a single-camera system with an angled camera configuration to compare the shape of a printed object with a CAD model, thereby identifying errors exceeding a specified threshold. In this setup, the camera is positioned at an angle relative to the printed object, enabling the capture of a two-dimensional image. To detect anomalies in the captured camera images, 3D models rendered with OpenSCAD are saved as STL files. Subsequently, an STL image is obtained by adjusting the viewpoint position in relation to the printed object. Camera calibration is conducted to describe the shape of the camera image, acquiring intrinsic parameters while mitigating distortion (Zhang, 2000). A grayscale-to-binary conversion is performed, representing a white background contrasting with black object forms. The same process is applied to export the shape of the STL image. Objects in both images are resized to a uniform size using edge detection techniques. Anomalies are identified by subtracting the STL image from the camera image, with a difference exceeding 5% indicating a print failure.

For multi-camera systems, Straub et al. (2015) proposed a multi-camera system for monitoring the 3D printing process from various angles. The system utilizes five cameras positioned around the 3D printer to capture images from different perspectives. These captured images are subsequently processed to extract relevant features, encompassing both geometric and photometric aspects. These extracted features play a crucial role in evaluating the quality of the printed objects. To detect print failures, a comparison is made between a pre-recorded video of a correctly printed object and the ongoing print process. Prior to monitoring the process for a specific object, it is necessary to record a video depicting the accurate printing procedure for that particular object. By comparing the real-time print process with the pre-recorded video, any deviations or discrepancies can be identified, indicating potential failures or errors during the printing process.



Figure 2.10 Multi-camera system experimental setup (Straub et al., 2015)

Likewise, in the system proposed by Nuchitprasitchai et al. (2017) et al., there is another setup for a multi-camera system. In this system, one camera serves to capture the overall printing process, while the second camera is positioned at a specific angle to record an alternative perspective of the printed object. The inclusion of multiple camera angles enhances the thorough inspection of printed objects, resulting in improved accuracy for defect detection. The proposed two-camera setup effectively detects printing errors by employing the Scale-Invariant Feature Transform (SIFT) and Random Sample Consensus (RANSAC) models to extract matching points from the images captured by the two cameras. By comparing the 3D printed objects with the 3D reconstructed images, errors can be accurately detected. The system has been tested with a variety of 3D objects using a difference threshold of 5%, showcasing its effectiveness in identifying printing errors. However, it is important to note that two-camera systems require increased computational resources due to the integration of images from two different angles. This higher computational demand must be considered during the implementation of the system. Additionally, the proposed dual-camera configuration relies on the SIFT and RANSAC models for matching points extraction from the camera images, enabling efficient detection of printing errors. Since the SIFT algorithm typically requires a substantial number of feature points, the matching process for a printed object with a simple shape may encounter errors due

to insufficient feature points. Similarly, the RANSAC models may introduce incorrect matches in cases where feature points exhibit similarities. These considerations highlight the potential challenges and limitations associated with the proposed dual-camera approach.

Single-camera systems offer the benefits of simplicity and real-time monitoring capabilities. Lyngby et al. successfully demonstrated the potential of their system in detecting printing failures, showcasing the effectiveness of single-camera setups. On the other hand, Nuchitprasitchai et al. improved defect detection accuracy by rendering printed models. The accuracy of the monitoring system is improved by image processing.

In contrast, multi-camera systems, such as the one proposed by Straub et al., provide a more comprehensive perspective and enhanced error detection capabilities. These systems capture images from multiple angles, enabling a thorough assessment of printed object quality in real-time by extracting features from the multi-angle images.

To summarize, both single-camera and multi-camera systems offer advantages and find applications in 3D printing. Single-camera systems excel in their simplicity and real-time monitoring capabilities, while multi-camera systems provide more comprehensive information and higher accuracy in detecting errors. It is important, however, to consider the limitations of these systems, which will be discussed in subsequent sections.

2.2.3 Summary of monitor systems

After an extensive review of the literature, it has been established that vision-based monitoring systems offer significant advantages in monitoring the quality of 3D printing. These systems leverage cameras and image processing algorithms to analyse the visual characteristics of printed objects, enabling the detection of defects such as cracks or gaps. Real-time feedback provided by these systems during the printing process allows for early error detection and improved reliability, particularly in industrial applications.

In addition, multi-camera systems can introduce computational complexity due to the requirement for more calculations. Moreover, the construction of multi-camera systems often entails higher costs and longer setup times. Taking these factors into consideration, the project opted to develop a vision-based surveillance system utilizing a single camera configuration. The objective is to enhance system stability and streamline the monitoring process, ultimately improving the accuracy and efficiency of detecting and resolving printing defects. By focusing on a single camera setup, the project aims to achieve a simplified and cost-effective solution for 3D printing quality assessment.

3. System Design

The system described in this thesis is designed to monitor the quality of 3D printing through the use of computer vision technology. The system is composed of four main parts. In the first part, the camera parameters are estimated to determine the camera's pose and parameters in world coordinates. The second part involves simulating the 3D printing process layer by layer using a slicing algorithm and a G-code simulator. The resulting simulated images are then used to assess the quality of the printed objects. In the third part of the system, computer vision algorithms and image processing techniques are used to extract features from the simulated and camera images. These include a background subtraction algorithm for segmenting printed objects, a thresholding algorithm for measuring infill density, and a Fourier-Mellin transform algorithm for evaluating shape features. The Fourier-Mellin transform was chosen as part of the computer vision algorithm in this system because some printed objects can have complex shapes that require advanced methods for accurate analysis. The Fourier-Mellin transform can detect shape-related defects, irregularities or deviations, an important aspect in ensuring that the final printed object meets the required quality standards. In the final part of the system, the extracted features are used to evaluate the quality of the 3D print and determine whether the printing process should be terminated. The details of this evaluation process are elaborated in the following chapter, while this chapter provides a detailed explanation of the first and second parts of the system. The system introduces an innovative approach to monitor and ensure the quality of 3D prints using computer vision technology. Figure 3.1 illustrates the overall process, and each of the components of the system is explained in greater detail in the subsequent sections.

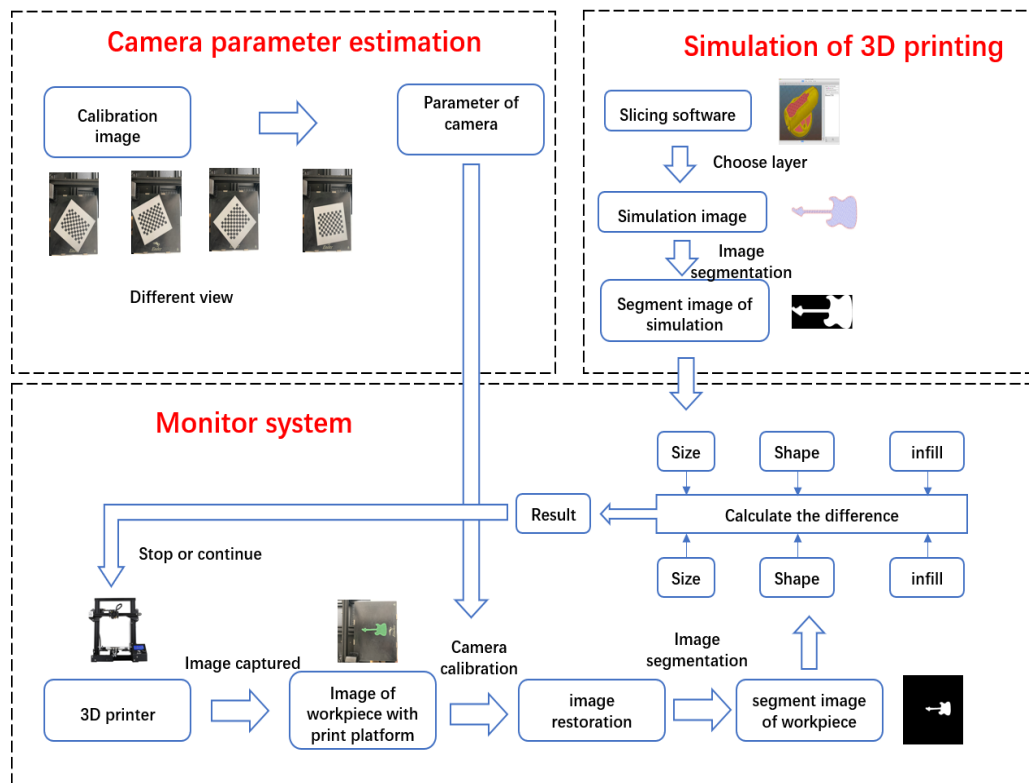


Figure 3.1: Flow diagram of proposed vision-based monitoring system

3.1 Camera parameter estimation

Camera calibration is a critical process in computer vision applications that utilize cameras to acquire images (Qi et al., 2010). It determines the relationship between points on an object's surface in space and their corresponding points on the image, as well as the camera's position relative to the object. This process estimates the camera's parameters and other factors that may affect image quality.

The first step in camera calibration involves capturing images of a known pattern or object, such as a checkerboard, and using them to compute the camera's intrinsic and extrinsic parameters. Intrinsic parameters refer to the internal characteristics of the camera itself. They are essential for understanding how light is projected onto the camera's sensor and how the sensor records this information. Key intrinsic parameters include:

- Focal Length (f): The focal length of the camera lens, which determines the

distance between the lens and the image sensor. It affects the field of view and the scale of the captured images.

- Principal Point (c_x, c_y): The coordinates of the principal point, which is the optical center of the camera's sensor. It represents the point where the optical axis intersects the sensor and is crucial for correcting image distortion.
- Lens Distortion Coefficients: Parameters that describe the distortion introduced by the camera lens, such as radial distortion and tangential distortion. Correcting lens distortion is vital for accurately mapping 3D points from 2D images.
- Pixel Size and Aspect Ratio: Information about the size of individual pixels on the camera sensor and the aspect ratio of the sensor.

Extrinsic parameters describe the camera's position and orientation in 3D space relative to the world or scene being captured. They are critical for accurately mapping 2D image coordinates to 3D world coordinates. Key extrinsic parameters include:

- Translation Vector (T): Specifies the camera's position in 3D space relative to a reference point, typically the camera's optical center.
- Rotation Matrix (R): Describes the camera's orientation in 3D space, often represented as a 3x3 matrix. It defines how the camera is rotated relative to a reference coordinate system.
- Extrinsic Matrix (M): Combines the translation vector and rotation matrix to represent the complete transformation from the camera's local coordinate system to the global 3D coordinate system.

To achieve camera calibration, the OpenCV library is used to capture multiple images of a checkerboard placed in different orientations above the printing platform and below the camera (Bradski et al., 2008). A checkerboard is used as a calibration object as it is a simple 2D pattern compared to more complex 3D

objects. To compensate for this limitation, the image of the checkerboard is captured through multiple viewing angles to obtain richer coordinate information.

As shown in the figure 3.2, the camera in the experimental setup is mounted on a bracket and located above the printing platform. During the entire printing process, the camera remains still and pointed down at the printing platform, allowing images and videos of the printing process to be captured.

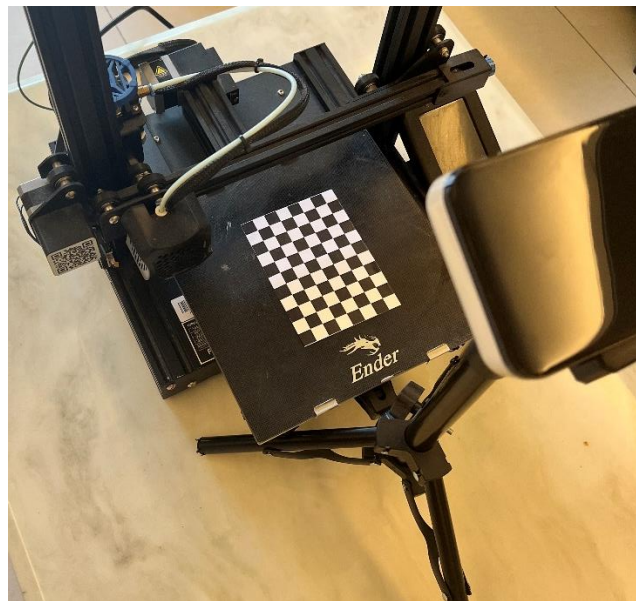


Figure 3.2: Illustration of the experimental printer and camera setup

To determine the position of the camera relative to the printing platform, several calibration images are captured from different viewing angles, and the calibration pattern is detected. The calibration images are shown in Figure 3.3, and a checkerboard of known size, consisting of 8x11 squares, each with a width and spacing of 11.5mm, is used to obtain a set of world coordinate points. The camera parameters can be estimated from the position of these points in the image. The next step in camera calibration is to refine the estimated intrinsic and extrinsic parameters by minimizing the reprojection error, which is the difference between the actual image points and the projected ones computed using the estimated parameters. This error measures the difference between the actual position of pattern points in the calibration image and the corresponding world point projected into the same image.

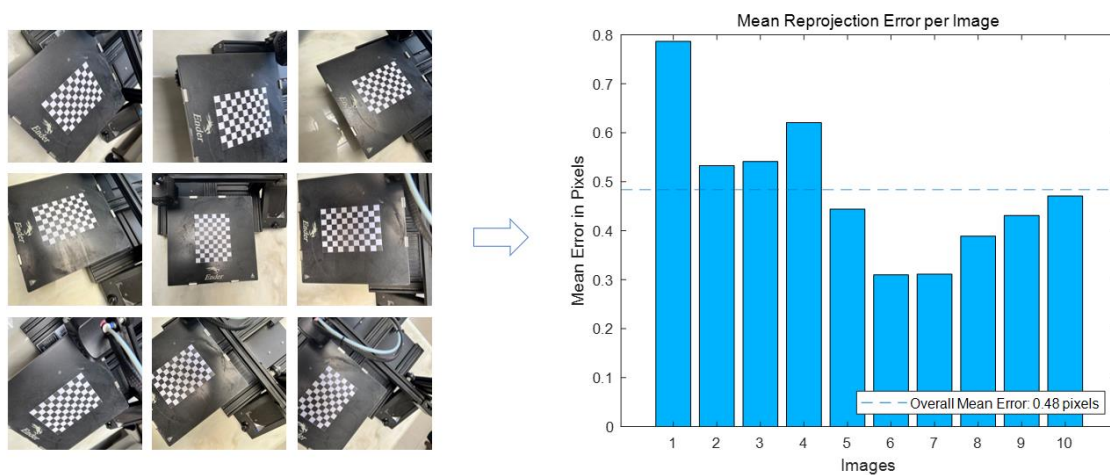


Figure 3.3: Mean reprojection error per Image

To determine the camera's position using the camera calibration parameters, the extrinsic parameters obtained during the calibration process need to be applied. These parameters include a rotation matrix and a translation vector, which describe the transformation from the camera coordinate system to the world coordinate system. The following transformations are applied to achieve this:

$$X_{world} = R \cdot X_{camera} + t \quad (3 - 1)$$

Where X_{world} is a 3D point in the world coordinate system, X_{camera} is the corresponding point in the camera coordinate system, R is the rotation matrix, and t is the translation vector.

Extrinsic parameters are used to convert a point in camera coordinates to a point in world coordinates. The camera's position in world coordinates can be calculated as the inverse of the rotation matrix multiplied by the negative of the translation vector:

$$C_{world} = -R' \cdot t \quad (3 - 2)$$

where C_{world} is the position of the camera in the world coordinate system.

The calibration process, which encompasses capturing images and processing them with the system, typically takes about ten minutes to complete. To ensure accurate calibration, specific lighting conditions are required: a light source should be positioned behind the camera in an environment with subdued lighting. It's important to note that recalibration may be necessary if there are changes in the ambient light conditions. This calibration approach serves a crucial purpose in the system. It allows for the determination of the printed object's position relative to the camera. Consequently, the system can accurately measure the size of the object positioned on the printing platform and provide the necessary camera parameters for generating simulated images based on the CAD model. Figure 3.4 illustrates the camera's relative position in relation to the printing platform. By adopting world coordinates that align with the print platform's coordinate system, the system can effectively replicate camera positions within the scene. This results in a more precise representation of objects within the camera's field of vision, enhancing the system's overall accuracy and reliability in assessing the quality of 3D prints.

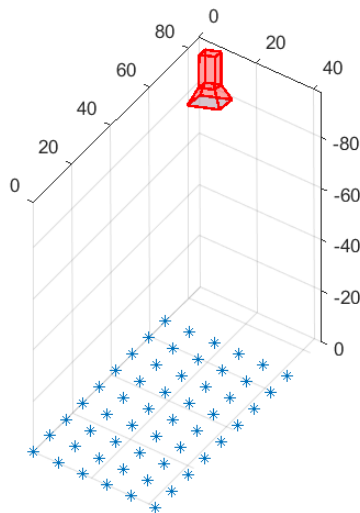


Figure 3.4: Position relative of camera and print platform

3.2 Print Simulation

A tool was developed for simulating the 3D printing process by rendering

individual layers of a CAD model. This tool was created to help assess the quality of the 3D printing process and to compare the simulated images with actual camera images of the printed layers. The process starts by importing a CAD model into the open-source slicing software, Slic3r, which generates the G-code needed for the 3D printer to execute the printing process. While the software can also generate simulated images, it cannot generate the specific type of lines required and cannot set where the camera is located. To overcome this limitation, a G-code simulator was developed specifically for this research. The simulator involves reading G-code and enabling it to generate simulated images via OpenGL. The generated G-codes were input into the simulator and executed line by line. The commands in the G-code describe the coordinates to be printed and the next coordinate to move to. Each pair of coordinates can be connected with a print line through the command from OpenGL library, and finally form a layer. Finally, the estimated camera parameters and the generated layer were used in conjunction with a pinhole camera model to render an image of the layer using perspective projection.

3.2.1 Slicing software

Slicing software is an essential tool in the 3D printing process, it plays a key role in converting a 3D model into a series of machine-readable instructions that a 3D printer can use to build the final object. The software takes a 3D model as an input, which can be in different file formats such as STL or OBJ, and then "slices" it into a series of 2D layers. These layers are then transformed into G-code or other machine-readable instructions that the 3D printer can understand and execute (Minetto et al., 2017).

The slicing process divides a 3D model into a series of 2D cross-sectional layers. The specific slicing algorithm used depends on the slicing software. The most common slicing technique for 3D models stored in STL is to take the triangular mesh of the 3D model and slice it along the Z axis to create a series of 2D slices. After the slicing process is complete, the software generates g-code, which contains instructions for the 3D printer to use to build the object. The g-code will contain information such as extruder position, hot-end temperature, print speed and layer height. The g-code also includes information regarding fill mode and

density, which determines how the internal structures of the object are filled. With the details present in the g-code, a g-code extractor was developed to simulate printing using the G-code produced by the slicing software.

3.2.2 G-code Extractor

The G-code, which is a programming language used to control the movements of a machine, is generated by inputting a digital model created by computer-aided design software (Baumann et al., 2017). Each line in the G-code file corresponds to a command and contains parameters to be executed (Latif et al., 2021). When there are no parameters input, the default operation will be performed. The most frequently used code is the G1 command, which is typically used to determine the toolpath for the extruder to move. In the G-code file, the first section consists of a combination of G and M commands that are used to prepare for printing. This section includes setting the temperature of the printer and nozzles, calibrating positions, moving units, and other important steps. A list of the most commonly used G-codes and M-codes used in 3D printing is shown in Table 3.1.

G-Code	Description
G1	Linear Move: The command is used as 'G1 X Y Z E F' where X is the position to move to on the X axis, Y is the position to move to on the Y axis, Z is the position to move to on the Z axis, E is the amount to extrude between the starting point and ending point and F is the federate per minute of the move between the starting point and ending point
G28	Move to Origin: This command is used to move specified axes to the endstops, if none are given, all of axes move backs away from each endstops. It's important to execute this command before printing due to it helps improve positional accuracy.

G92	Set Position: This command allows resetting the current position to the values specified. It can also be used to change the amount to extruder and federate.
M24	Start / resume print
G90/G91	These commands are used to switch between absolute and relative positioning. In absolute positioning, the extruder moves to a specific position on the build platform, while in relative positioning, the extruder moves a specified distance from its current position.
M104	This command is used to set the temperature of the hotend.
M140	This command is used to set the temperature of the build platform.
M190	This command is used to wait until the build platform has reached the desired temperature before starting the print.
M220	This command is used to set the speed multiplier for the extruder movements.

Table 3.1: Most common G-Codes/M-Codes used in 3D printing.

Listing 3.2 illustrates the use of M-codes to prepare the 3D printer for printing. This includes setting the bed and extruder to their target temperatures and calibrating the extruder's position to the end stops. The G1 command defines the current and next printing coordinates, with each pair of coordinates connected by a print line. This forms the layers of the printed object.

- 1 M107 ; fan off
- 2 M190 S65 ; set bed temperature and wait for it to be reached

```
3 M104 S185 ; set temperature
4 G28 ; home all axes
5 G1 Z5 F5000 ; lift nozzle
6 M109 S185 ; set temperature and wait for it to be reached
7 G21 ; set units to millimeters
8 G90 ; use absolute coordinates
9 M82 ; use absolute distances for extrusion
10 G92 E0
11 G1 Z0.350 F7800.000
12 G1 E-2.00000 F2400.00000
13 G92 E0
14 G1 X81.852 Y44.273 F7800.000
15 G1 E2.00000 F2400.00000
16 G1 F1800
17 G1 X84.948 Y41.029 E2.40773
18 G1 X87.362 Y39.420 E2.67152
```

Listing 3.2: A part of example G-code program to print an object.

Extracting positional parameters is an important step in simulating printing. The G1 command determines where the alignment head will move next. Because the process of 3D printing is the process of moving and stacking materials. Therefore, the image of each layer can be obtained by simulating the moving path of the alignment head (Jin et al., 2014). The algorithm describing the process of extracting parameters is shown below.

ALGORITHM 2: G-code Extractor

```
// This function is used to extract the coordinates of G-code
```

```
Function [x, y, z] = Extract_Gcode (G-code)
```

```
//The program will read line of G-code until it's empty
```

```
while Line Is Not Empty, do
```

```
i equal to 1, i ++
```

```
if Line equal to G1 command
    g1_command = Line[i]
    //Read current coordinate
    [x[i],y[i],z[i] ]= Read Line(g1_command)
    CurrentCoordinate.x = x[i]
    CurrentCoordinate.y = y[i]
    CurrentCoordinate.z = z[i]
    end
end
end
```

3.2.3 Image simulation

The image simulation system employs Three.js, a third-party JavaScript (JS) library for WebGL, to generate simulated images. Three.js offers various functions to render 3D scenes on web browsers, such as creating objects with different materials, colors, and geometries (Dirksen, 2013). To render the scene, the system includes a camera, renderer, and scene components. The light source for rendering can be chosen from ambient light, directional light, or point light source. In this experiment, ambient light is used to avoid any noise resulting from the absence of lighting. The camera's position is determined by the camera calibration result, and perspective projection is set to render the scene. Lastly, a renderer is created to integrate the scene and camera components. A diagram of the Three.js system is provided below.

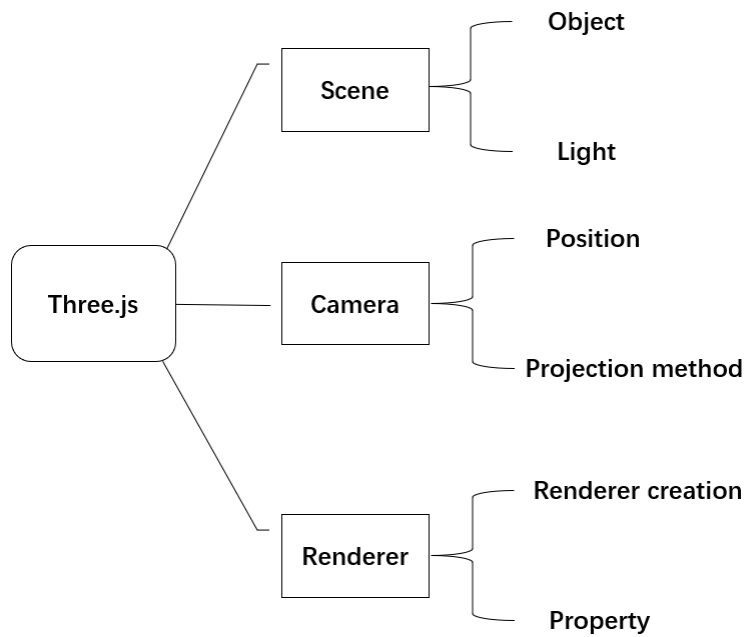


Figure 3.5: Overview of three.js system

To generate simulated images using the system, a series of steps are followed. Firstly, the Three.js library is utilized to create a 3D scene, which includes objects, materials, and lighting. The parameters for creating objects can be extracted from the g-code extractor. Next, the camera's position and projection are determined based on calibration results. Following this, a renderer is created to combine the scene and camera components. The renderer then produces a 2D image of the 3D scene from the camera's viewpoint, which is displayed on the web browser. For rendering the scene, ambient light is chosen as the lighting condition to create suitable simulated images for the objects in the scene. The simulated image generated is presented in Figure 3.6, which also displays the material and lighting used for the object.

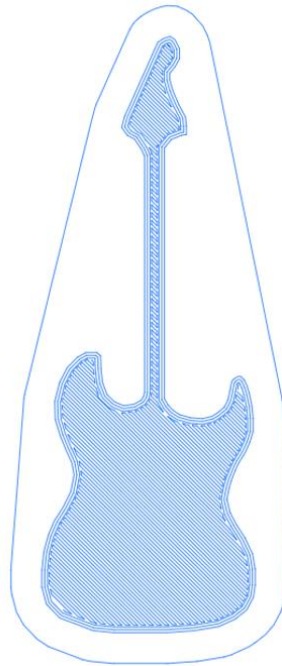


Figure 3.6: Simulated Image with Ambient Lighting for Printed Object.

Figure 3.7 shows the simulation results of the complete printed object from different angles, which can be used to check the integrity of the final printed object. In order to make the position of the camera relative to the print object consistent with the real situation, all print points are recalculated to align with the print coordinate system. Then, lines are created based on the definitions of transformed point and updated into the scene, the object is formed after all of lines are added.

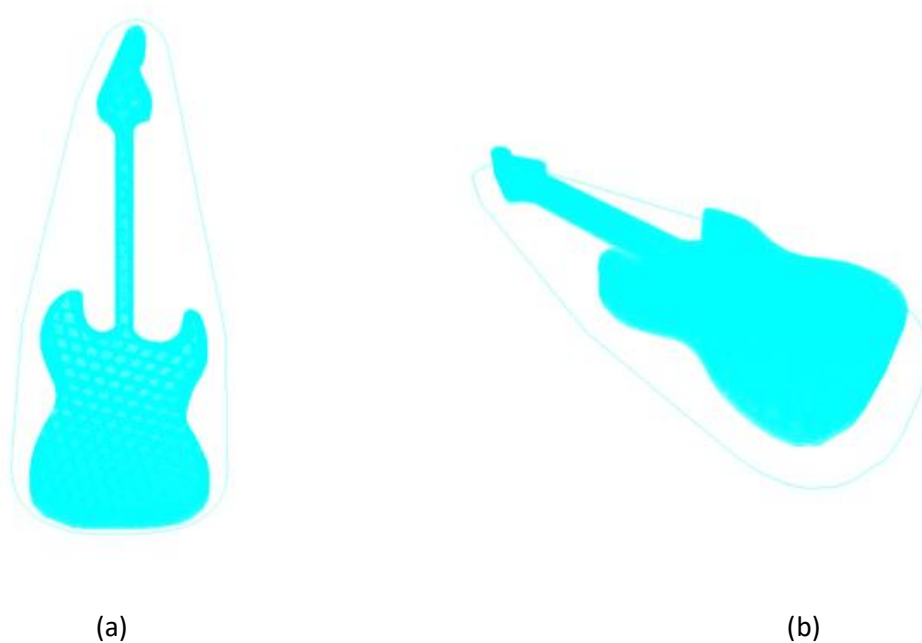


Figure 3.7: Simulated object. (a) simulated 3D object from top view (b) simulated 3D object from an example angle

3.3 Feature extraction

This section provides an overview of the methods used for feature extraction, which involves identifying and extracting relevant features such as shape, size, and infill density from the raw data of printed objects. Extracting these features helps assess print quality, detect anomalies, and optimize the printing process. This section delivers an exhaustive overview of various techniques and methodologies employed for extracting significant data from 3D printing data. To extract the shape of the printed object in the image, the background is removed using background subtraction. Thresholding is applied to capture fill material in printed objects. The evaluation of shape features from the frequency domain is done using the Fourier-Mellin transform. Owing to material variations inherent in the 3D printing process, it's common for printed 3D components to deviate from their CAD file representations in terms of shape and size. This divergence can be seen as a consistent offset, making it pertinent to consider when establishing thresholds for identifying failures.

3.3.1 Background Subtraction

Background subtraction is a widely used technique in computer vision and image processing for extracting the foreground objects or features in an image by removing the background (Nurhadiyatna, 2013). This technique is a basic step for feature extraction in image analysis, as it allows for the isolation of the foreground objects, which is useful for extracting the infill pattern in later section that will be described.

The process of background subtraction involves examining the disparities between a current image and a reference image. By subtracting the reference image from the current image, an output image is generated that accentuates the areas where foreground objects are present, while the background regions exhibit consistent and uniform values. In this project, several general steps are followed to implement background subtraction, including acquiring a reference image, subtracting it from the current image, and subsequently applying morphological operations. Figure 3.8 visually illustrates the extraction of foreground components from two images. In this context, the reference image represents the printing platform without any printing, while the other image corresponds to a specific frame obtained after a particular layer has been printed during the printing process. By applying morphological operations to the result of the background subtraction of these two images, a foreground mask can be derived. This foreground mask serves as a tool to isolate the foreground objects in the image.

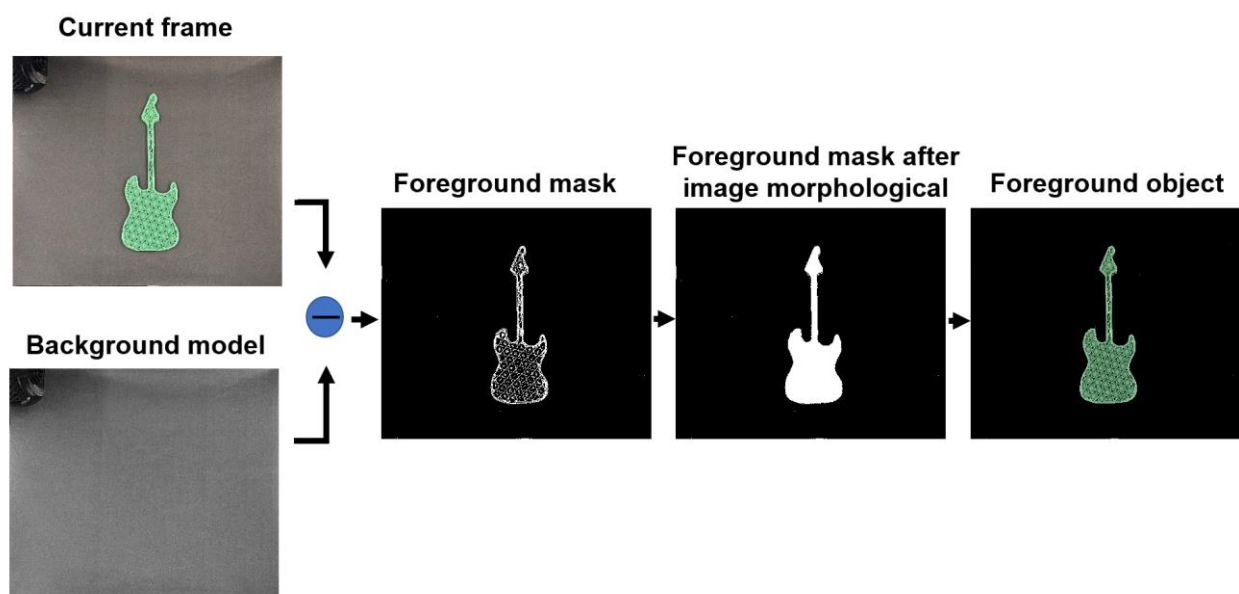


Figure 3.8: Flowchart of foreground extraction.
Development of a Vision-based Monitoring System for Quality Assessment

In addition, the binary image produced by the background subtraction also contains the contours of the objects and the background is represented as black. This binary image, also known as the foreground mask, provides valuable information about the object, such as its shape and size. This information is then used by the system to analyse and compare the printed object with the simulated object. It is important to note that the background subtraction is only applied to the camera image and not the simulated image as the latter already has the printed object and background separated into two different colours during the generation of the G-Code simulation.

After the foreground extraction operation, the extracted objects are ready for feature extraction processing. The feature extraction process includes size and infill pattern. These features allow the system to evaluate later.

In conclusion, the background subtraction method is a crucial step in the process of evaluating the printed object's features. By accurately extracting the object from the background, the system is able to obtain meaningful information about the object, which can be used for quality control and improvement purposes.

3.3.2 Thresholding for infill density

Thresholding is a commonly used image segmentation technique that separates the target object from the background (Opencv, 2019). This method will be used in this project to extract infill density features from the results of foreground extraction (Sezgin et al., 2004).

The method of estimating infill density using images taken from the top can be considered a limitation in certain contexts. While it is a viable approach and can provide valuable information about infill density, it may have limitations in accurately representing the true internal structure of the printed object. For example, the quality of the images and the accuracy of the thresholding method can be sensitive to lighting conditions and the presence of shadows. Variations in lighting may affect the accuracy of infill density estimation. Besides, objects with complex internal geometries or intricate structures may not be accurately represented through top-down images alone.

Therefore, while estimating filling density from top-view images is a useful and practical approach, it should be applied with awareness of its limitations.

To apply the thresholding technique, the images need to be converted to grayscale, as the algorithm operates on grayscale images rather than colour ones. This involves transforming the three-dimensional arrays that represent colour images into one-dimensional arrays, where each value in the array represents a pixel with a value ranging from 0 (black) to 1 (white), instead of the RGB's three-valued pixel. By doing this, the image retains its contrast information while losing its colour information (Swzgin & Sankur, 2004).

Next, the foreground image, obtained through background subtraction and representing the printed object, undergoes a thresholding process to extract the infill material. This step focuses on effectively separating the infill material from the target object by applying an appropriate threshold value. Given that the density of the printed object is already known, multiple thresholds can be applied to the image and determine the most suitable threshold. After applying the thresholding, the density of the printed object can be calculated using the following formula:

$$D = T_w / T_p \quad (3 - 3)$$

Here, T_p represents the total number of pixels of the printed object, while T_w represents the total number of white pixels within the printed object. In this context, the white pixels correspond to the filling material portion of the object. By calculating the ratio of white pixels to the total number of pixels, the density of the printed object can be obtained. Figure 3.9 illustrates the relationship between different threshold values and the calculated infill density. The figure demonstrates that when the threshold is set to 0.69, the resulting calculated infill density is closest to 20% of the original density of the printed object. Additionally, 10 repeated experiments were conducted under consistent lighting conditions and using the same printing material. The average threshold value obtained from these experiments was 0.65. Hence, for this specific experimental setup, the threshold value of 0.65 can be utilized. However,

it is important to note that if there are any changes in the experimental conditions, the threshold value should be recalibrated accordingly.

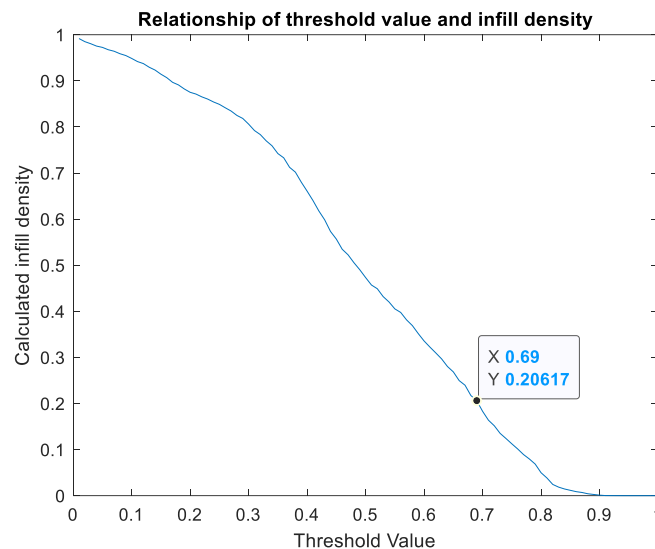


Figure 3.9: Relationship of threshold value and infill density.

Figure 3.10 shows the resulting image after thresholding using the adjusted threshold value. The thresholding operation is specifically applied to the foreground image. In the resulting output, the white region corresponds to the filling material of the printed object, while the black region represents the void or empty space. The result of thresholding will be utilized for subsequent assessments of the density of the printed objects.

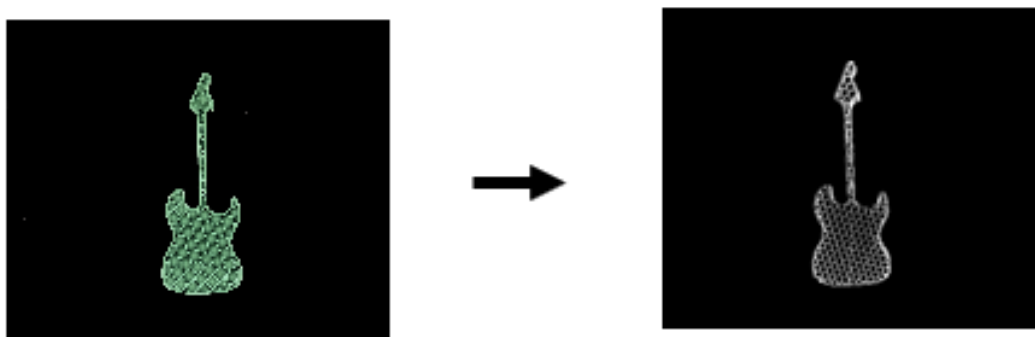


Figure 3.10: Result of thresholding.

3.3.3 Fourier-Mellin Transform

Fourier-Mellin transform (FMT) is based on Fourier transform analysis, which converts the images to erase the effects of rotation, scaling and translation (RST) (Kazik & Göktoğan, 2011). The Fourier-Merlin transform approach combines Fourier and Merlin transforms with rectangular-to-polar transformations, yielding a computationally efficient RST-invariant matching scheme. In this project, the FMT is employed to assess the shape of printed objects. By converting the shape of the printed object from the time domain to the frequency domain, it becomes feasible to evaluate the shape in terms of frequency.

In this project, the Fourier-Merlin transform consists of four processing steps. They are Fourier transform, rectangular coordinates to polar coordinates conversion, logarithmic scaling and second Fourier transform. The first Fourier transform is to achieve translation invariance. rectangular coordinates to polar coordinates Conversion Converts rectangular coordinates to polar form. In this case, any rotation in rectangular coordinates appears as a horizontal movement in polar coordinates, and any scaling in rectangular coordinates appears as a vertical offset in polar coordinates. Then, the Merlin transform is implemented using logarithmic scaling and second Fourier transform. Mellin transforms are used to implement scaling invariants. Figure 3.11 below shows the flowchart of Fourier-Mellin Transform method. In this figure, $f(x, y)$ is the image signal, $|F(u, v)|$ is its Fourier Transform. $F(r, \theta)$ is the transformation of $|F(u, v)|$ from rectangular coordinates to polar coordinates. Through the logarithmic scaling of the polar coordinates, the obtained $F(e^\rho, \theta)$ can be used for Fourier Mellin transform to obtain $|M(u, v)|$ that can be used for matching.

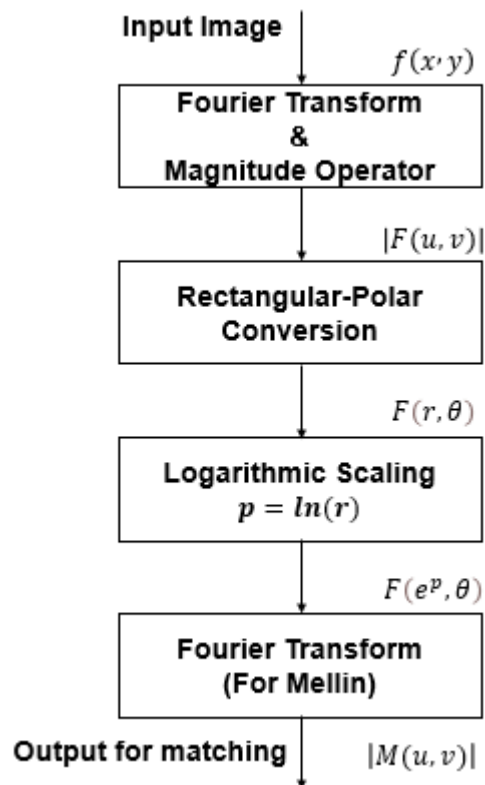


Figure 3.11 Sequential processing in Fourier-Mellin Transform

The Fourier-Mellin transform method consists of the four steps shown in the figure. The combination of these steps will achieve the RST transform invariant. These steps will be described in detail below. Basically, the last two steps are used to implement the Mellin transform, so they would be introduced together.

The Fourier transform converts a time-dependent signal from the time domain to the frequency domain. When the signal is space-dependent, it would be transformed from spatial domain to frequency domain. The expression of Fourier Transform is given by:

$$F(\omega) = \int_{-\infty}^{+\infty} f(t) \cdot e^{-j\omega t} dt \quad (3-4)$$

Where $F(\omega)$ is the Fourier Transform of signal $f(t)$. According to the property of Fourier Transform, if a signal has been shifted in time domain, the relationship between its corresponding Fourier Transform and the magnitude of Fourier Transform corresponding to the original signal is given as:

$$F_{\tau}(w) = F(w) \cdot e^{j\omega\tau} \quad (3-5)$$

Then, the expression could be rewritten as:

$$F_{\tau}(w) = |F(w)| \cdot e^{j\omega\theta} \cdot e^{j\omega\tau} = |F(w)| \cdot e^{j\omega(\theta+\tau)} \quad (3-6)$$

Where θ is the phase angle of the Fourier Transform. Therefore, the difference between the original signal and the time-shifted signal only exists in phase component. Their magnitude components are invariant for a given time shift:

$$|F_{\tau}(w)| = |F(f(t - \tau))| = |F(w)| \quad (3-7)$$

The detail example of the time shifting is show below.

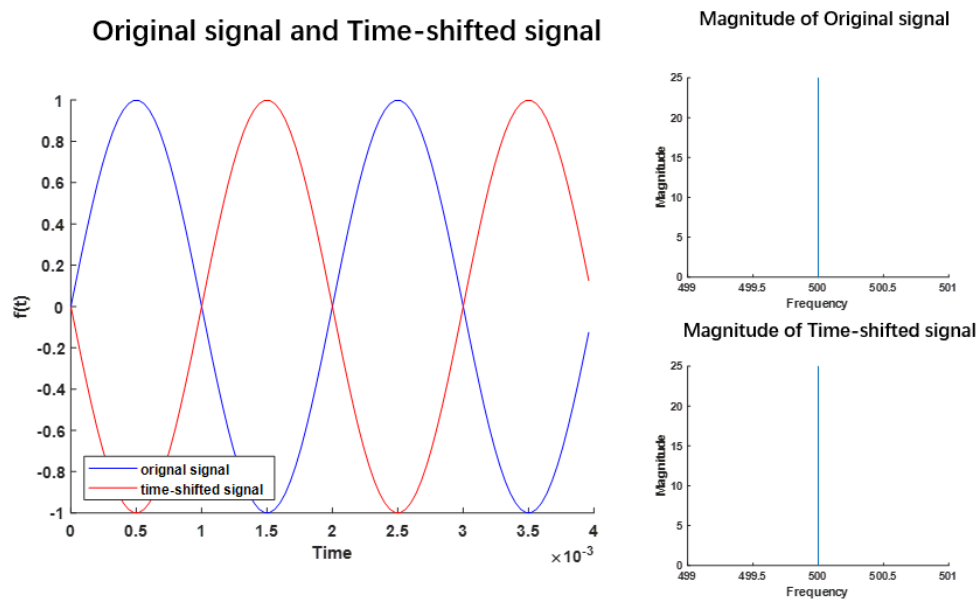


Figure 3.12 An example of translation invariant in Fourier Transform

In above diagram, a one-dimensional signal with a frequency of 500 Hz is generated and show on the graph in blue. The other is a time-shifted signal with the same frequency and shown in red on the graph. After the Fourier Transform, it is apparent their magnitudes of Four Transform are the same. Thus, the Fourier Transform can

achieve time shifting invariant for one-dimensional signal by ignoring the phase information.

However, all the signals are processed as digital signal in computer which means that the signals are discrete. Hence, the Discrete Fourier Transform is used to analysis discrete signal which can be expressed as:

$$X(u) = \sum_{n=0}^{N-1} x(n) \cdot e^{-\frac{j2\pi i}{N}} \quad u = 0,1,2, \dots, N - 1 \quad (3 - 8)$$

In the digital image processing, the image signal is two-dimensional signal. Hence, the two-dimensional Discrete Fourier Transform has been addressed as

$$F(u, v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \cdot e^{-j2\pi \left(\frac{ux}{M} + \frac{vy}{N} \right)} \quad u = 0,1,2, \dots, M - 1 \quad v = 0,1,2, \dots, N - 1 \quad (3 - 9)$$

Where $f(x, y)$ is the image signal, $F(u, v)$ is its Fourier Transform. For one-dimensional signals, the Fourier transform can be used to achieve time-shift invariants. From the two-dimensional signal, the Fourier transform can be used to realize the translation invariant, which is the same as the one-dimensional time-shift invariant. Also, the rotation of the image function will rotate the magnitude of its Fourier transform by the same amount, and the scale factor a in the image will become $1/a$ of the magnitude of its Fourier transform.

In the rectangular coordinates to polar coordinates conversion of image $f(x, y)$, the effects of rotation and scaling effects of shapes in the image are separated, that is, from (x, y) coordinates to (r, θ) coordinates in Figure 3.13.

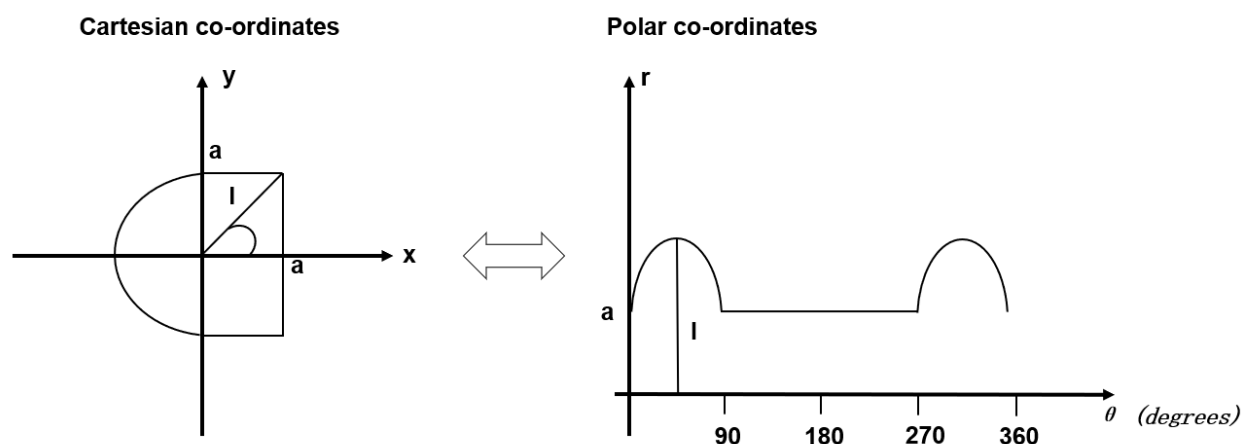


Figure 3.13 An example of rectangular coordinates to polar coordinates conversion

In above diagram, the right coordinates represent the polar coordinate which is another representation of the image and contains the same information as the left one. Any rotation of $f(x, y)$ behaves as a shift of θ in $f(r, \theta)$. A change in scale of $f(x, y)$ only affects the r coordinate of $f(r, \theta)$. Hence, a two-dimensional scaling of the image function is reduced to the polar representation of $f(r, \theta)$ which represent a scaling of only one coordinate (Ho & Goecke, 2008).

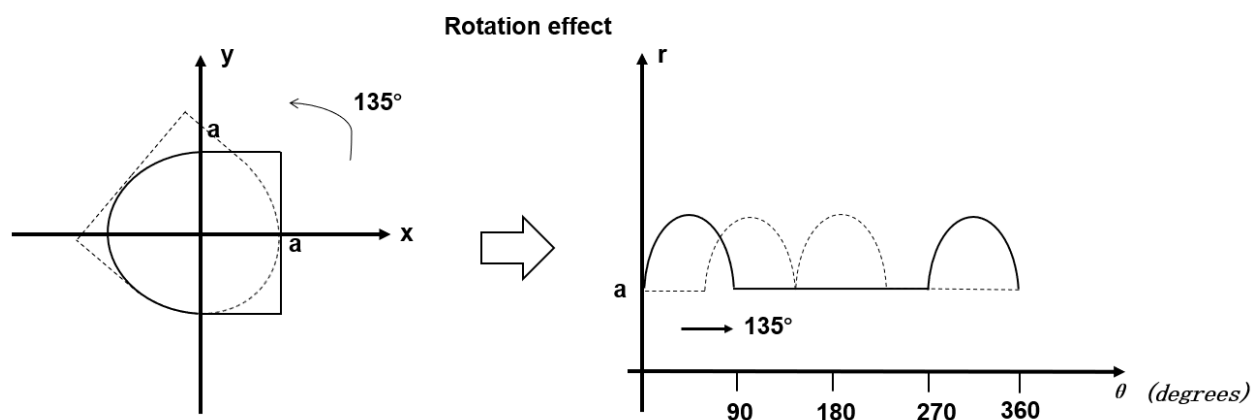


Figure 3.14 An example of rotation effect

As shown in Figure 3.14, rotation of a shape in Cartesian coordinates will appear as a horizontal movement in polar coordinates over the range of angles. According to this situation, whether there is a rotation between the template and the object in the image can be obtained by detecting whether it matches the polar coordinate image of the object after shifting the polar coordinate image of the template.

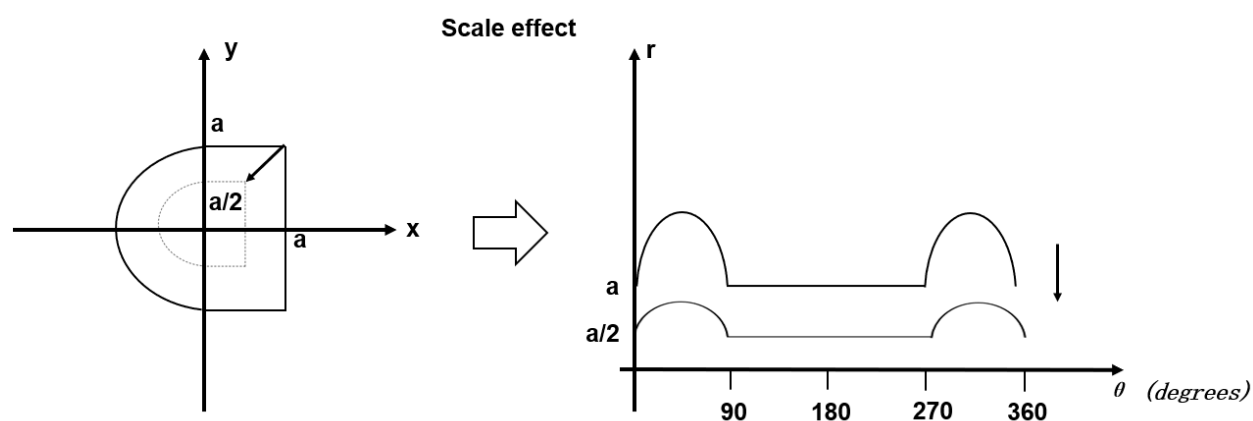


Figure 3.15 An example of scale effect

Similarly, the scaling effect of objects in the image will be represented by vertical shifting in polar coordinates. In other words, the shape on polar coordinates shrinks vertically on a logarithmic scale. Since the rotation and scaling in the image becomes a problem of horizontal and vertical movement respectively in polar form. Therefore, checking for rotation or scaling between the template and the object is simplified in shape comparison.

The last step is to perform a Fourier transform on the result obtained in the above steps and then phase correlate. The resulting image in the transform domain is the result of the Fourier-Mellin Transform, this method has strong robust to scaling and rotation of images. The Fourier-Mellin Transform of an image function $f(r, \theta)$ is given by:

$$M_f(u, v) = \frac{1}{2\pi} \int_0^\infty \int_0^{2\pi} f(r, \theta) r^{-ju} e^{-jv\theta} d\theta \frac{dr}{r} \quad (3-10)$$

where the Fourier integral is seen to be combined with the Mellin integral with the former applied along the angular direction and the latter along the radial direction. In order to see its scaling and rotation invariance, a function $f_2(\alpha r, \theta + \beta)$ is obtained from scaling and rotate the image function $f_1(r, \theta)$. Their Fourier-Mellin Transform are related as follows:

$$M_{f_1}(u, v) = \alpha^{-ju} e^{-jv\beta} M_{f_2}(u, v) \quad (3 - 11)$$

where the magnitudes of $M_{f_1}(u, v)$ and $M_{f_2}(u, v)$ have a translation in the r and θ axes. By substituting $r = e^\rho$, the FMT can be expressed as a Fourier transformation:

$$M_f(u, v) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_0^{2\pi} f(e^\rho, \theta) r^{-ju\rho} e^{-jv\theta} d\theta d\rho \quad (3 - 12)$$

Hence, applying the Fourier-Mellin Transform to input image can achieve the scaling and rotation invariance. Besides, after remapping the transformed spectrum to log-polar coordinates, a new representation of translation, rotation and scale invariants is created.

As mentioned before, Fourier-Mellin Transform is a combination method to achieve the RST transformation in shape comparison. According to the properties of Fourier-Mellin Transform, it transforms the image function from the spatial domain to the frequency domain. Besides, Fourier-Mellin Transform is still invariant to rotation, scaling and translate of the shape in shape comparison. An example is given in Figure 3.16, the translated image is obtained by translating the shape in the original image. After applying the Fourier Transform to both images, it is clear to find their magnitudes of Fourier Transform are the same in the response image. This indicates that the Fourier Transform is translation invariant.

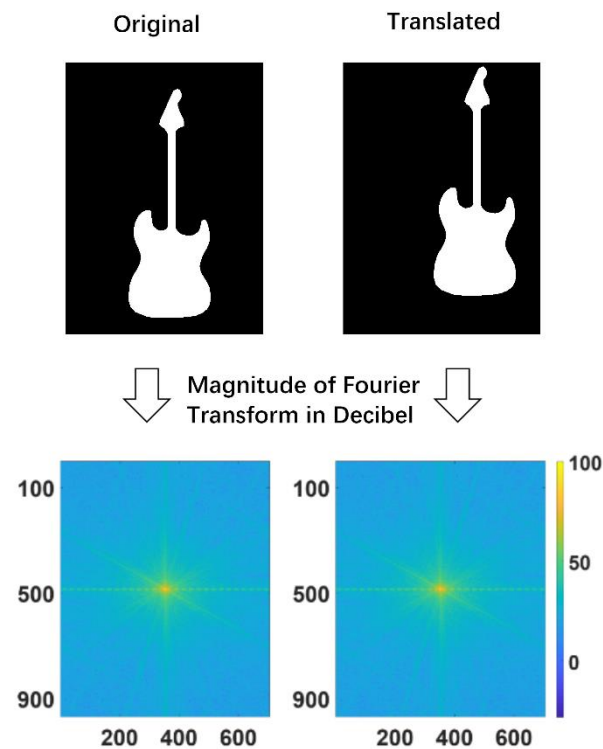


Figure 3.16 An example of translation invariant of Fourier Transform

Rotation invariant and scaling invariant are important properties in the Fourier-Mellin Transform. The process of implementing rotation invariants consists of the following steps. After the first Fourier transform, the rotation of the image function rotates the magnitude of the Fourier transform by the same amount, and then a rectangular coordinate to polar coordinates transformation converts the magnitude of the Fourier transformed image to polar form. Therefore, any rotation of the image function appears as a horizontal movement in polar coordinates, that is, a horizontal translation in polar coordinates. In this case, the rotation problem has been transformed into a translation problem in polar form. Finally, this special "rotational" invariant can be achieved by the second Fourier transform; because the Fourier transform can achieve the translation invariance as mentioned before. Scaling invariants can be achieved by the Mellin transform. A common implementation is a logarithmic scaling of the image coordinates, followed by a Fourier transform. The logarithmic scaling has been achieved in the rectangular coordinates to polar coordinates conversion. From the perspective of the implementation of the scaling invariant, it is very similar to the implementation of the rotation invariant. After the first Fourier transform, the scale change of the image function by factor a becomes $1/a$, and then the rectangular coordinates to polar

coordinates transformation transforms the magnitude of the Fourier transformed image into polar form. Therefore, any scaling of the image function moves vertically in polar coordinates, that is, translates vertically in polar coordinates. In this case, the scaling problem becomes a translation problem in polar coordinates. Finally, this special "scaling" invariant can be achieved by a second Fourier transform. It should be noted that the sampling method of the scaling axis in the rectangular coordinates to polar coordinates transformation is not linear, but logarithmic, that is, linear scaling in the rectangle will become logarithmic vertical movement. Figure 3.17 shows an illustration of Fourier-Mellin Transform, one of image is obtained by rotating and scaling the original image. After applying the Fourier transform to both image and remapping the transformed spectrum to log-polar coordinates, a new representation of translation, rotation and scaling invariant is created (Goodman, 2005). After the first Fourier transform, the relative translation of the objects in the two images is eliminated, while the relative rotation and scaling are passed as translations after remapping the transformed spectrum to log-polar coordinates. The translation in the log-polar domain is then decoupled by applying a second Fourier transform. It can be seen that the resulting images are almost identical without the translation, rotation and scaling changes of the same object. In this study, FMT was used not only to extract features to compare the contours of printed layers, but also to extract features to evaluate the infill patterns of printed layers.

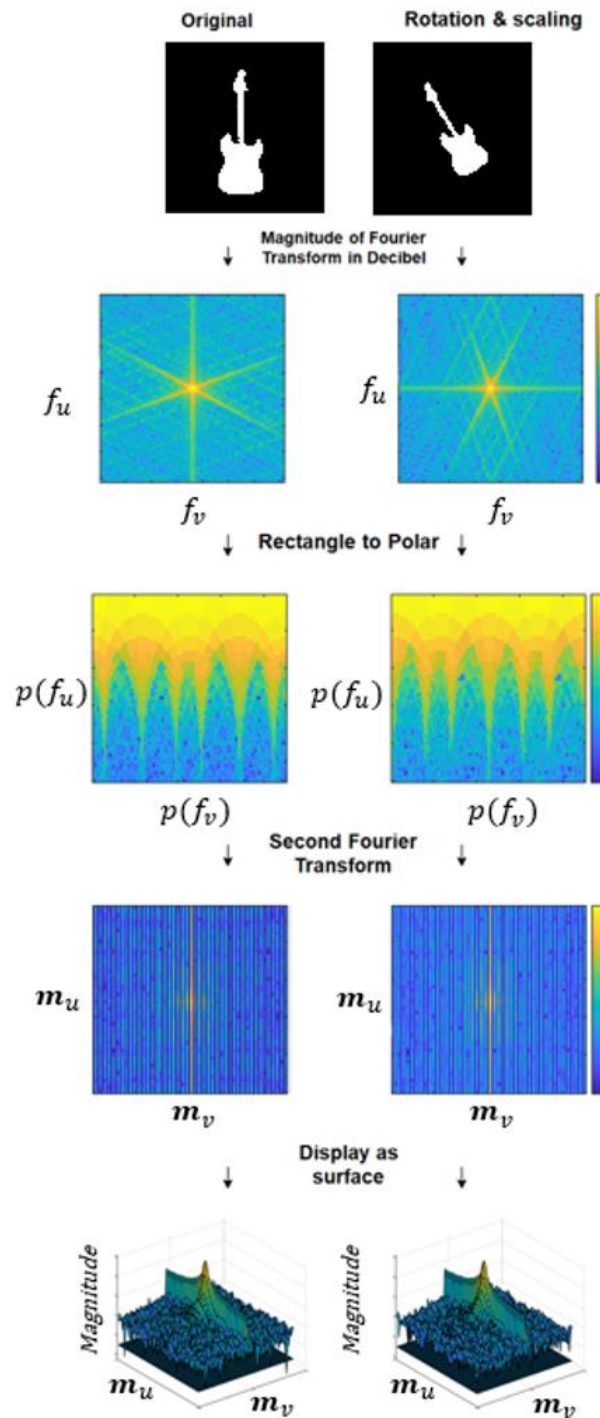


Figure 3.17 An illustration of Fourier-Mellin transform

4. Feature Evaluation

Feature Evaluation will evaluate print quality from three proposed features. These three features can be measured on the captured image and simulated image through the methods described before. These three features are the size of the printed object, the density of the infill layer and the shape of each layer. Among these, the size of the printed object can be measured by camera parameter estimation and background subtraction method, the density of the infill layer can be measured by foreground extraction and image thresholding method and the shape of each layer can be measured by FMT method. These details will be described next.

4.1 Evaluation of size

The size of the printed image will be the first detected feature to help detect other features. First, background segmentation is performed on the image of the printed object to be processed. This step will convert the background pixels of image to black leaving only the part of the printed layer. Then, the image is converted into a binary image to remove redundant information and only keep shape information. This will improve the efficiency of calculating the size of printed objects.

In this experiment, a bounding box is used to measure the size of an object. The bounding box generally refer to the coordinates of border enclose image object. Then, the size of object can be calculated by the width and length of the border. To find the bounding box of image object, blob analysis is used to find the connected component in the image. Since the image target is binary image, the connected component of image object can be detected by finding the connected region of white pixels. Finally, the bounding box can be found by calculating the minimum area circumscribed rectangle of the connected region.

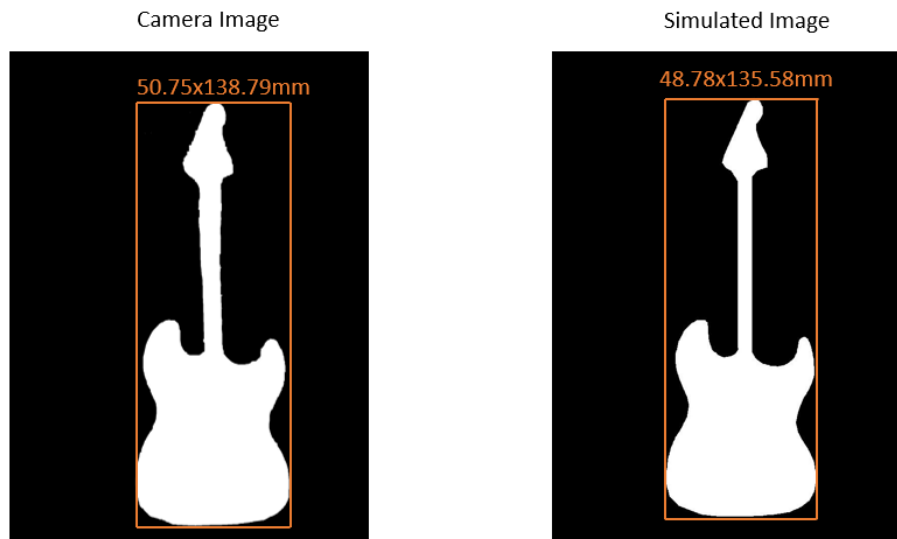


Figure 4.1 Size Instruct of camera image and simulated image through bounding box

As shown in Figure 4.1, in the camera image, the shape of the printed object is drawn on a black background. In the simulated image, the shape of the printed object is drawn in the same image as the same algorithm was applied to. Although the contours in the camera image closely match the shapes in the simulated image, there are still slight deviations in some places which will affect the evaluation of features. In the slicing software, the dimension of simulated image is known in advance as it is settled before printing. Then, the size represented by each pixel can be known by the camera calibration method described previously. Next, the dimension of printed object can be measured by counting the number of pixels contained in the length and width of the bounding box. Finally, the error of size can be measured by comparing the size of camera image and simulated image.

The significant error of 2*3mm between the camera image and simulated images, as depicted in Figure 4.1, can be attributed to several factors. First, despite the calibration process described earlier, which aims to establish an accurate relationship between pixels in the camera image and real-world dimensions, there can still be residual calibration errors. These errors may result from inaccuracies in the camera's intrinsic and extrinsic parameters estimation, lens distortions, or other factors. Even minor errors in calibration can accumulate and lead to noticeable discrepancies in object size measurements. Besides, the resolution of the camera sensor and the number of pixels

available for capturing the object in the camera image can affect measurement accuracy. If the camera has a relatively low resolution or pixel count, it can result in less precise size measurements. Moreover, lighting conditions during image capture can introduce variations in object appearance, leading to minor deviations in the contours and edges of the printed object. Shadows, reflections, or uneven lighting can affect the perceived shape and size of the object. Last, Optical distortions inherent to the camera lens or sensor can lead to inaccuracies in the captured image. Therefore, a tolerance of five percent is given on the dimensions to minimize errors related to these factors.

4.2 Evaluation of infill density

The next step is to evaluate the infill density of the printed layer. By employing the methods described in section 3.3.2, the images containing infill materials and their corresponding infill densities have been calculated. To evaluate the effectiveness of this approach, the same operations will be applied to simulated images in order to compare the results.

Figure 4.2 depicts the process of calculating the infill density for the simulated image. In this illustration, the white pixels represent the infill material, the red pixels represent the total area, and the calculated infill density is determined to be 20.57%. This value closely aligns with the infill density previously calculated for the printed object. The expected printed object infill density is 20%, and the previously calculated infill density for the printed object is about 20.62%, which has an error of no more than 5%. Thus, this method effectively demonstrates the validity of the approach and the ability to evaluate the density of the infill material.

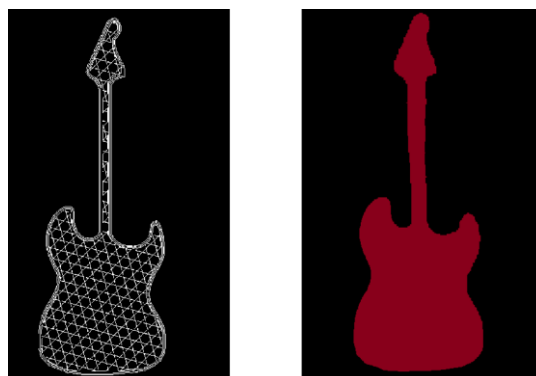


Figure 4.2 An example of the density calculation

4.3 Evaluation of shape

The last step is to evaluate the shape of the printed layer. This feature will be evaluated by calculating the 2D correlation coefficient between their resulting FMT images. As described before, FMT can perform Fourier transform to the input image and remap it to log-polar coordinates. Therefore, the difference between the simulated image and camera image can be examined by 2D correlation coefficient.

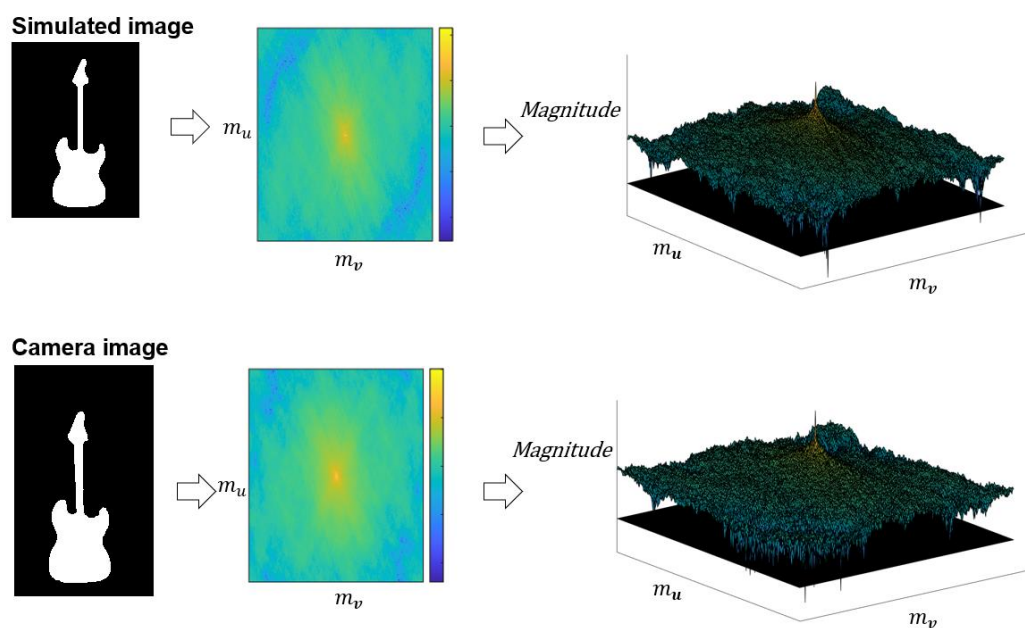


Figure 4.3 FMT images of the camera and simulate image

As shown in Figure 4.3, In the figure, it can be seen that the FMT images of the camera and simulate images are almost identical for the printed layer of a guitar and their correlation coefficient is 0.9931.

5. Experimental Analysis

The 3D printed monitoring system was tested to evaluate its ability to accurately track and measure the dimensions and patterns of objects of different shapes and sizes. The test involved printing a range of objects using plastic materials, including cubes, cuboids and cylinders. The results showed that the monitoring system was able to capture the size of the printed object, regardless of its shape, within a 5% error. This demonstrates the system can monitor, and measure 3D printed objects in the given condition, which is critical for ensuring quality control and precision during manufacturing.

5.1 Test on different shape

Due to the limitations in the camera calibration and feature extraction procedures, there are imperfections can stem from inaccuracies in estimating camera intrinsic and extrinsic parameters, lens distortions, or other related factors. Therefore, there are inherent displacements between the features extracted from the camera image of the printed layer and the corresponding simulated image created via G-code, which need to be tolerated and excluded from the detection of printing errors in the quality assessment. To investigate the range of intrinsic displacements and establish tolerances for quality monitoring and process control, a series of experiments were conducted using a Creality Ender 3 3D printer. Six objects, encompassing regular and irregular shapes, were printed, with the results analysed based on layer 10 for each object. These objects are outlined in Table 1. Through the analysis of extreme displacement values observed in the experiments, the proposed criteria outlined in Table 1 indicate that for achieving high-quality 3D printing, the allowable tolerance between the camera image and the simulated image should fall within $\pm 5\%$ deviation in terms of size and infill density. Additionally, a correlation coefficient exceeding 0.88 should be maintained.

The shape of the objects being printed in 3D can have a significant impact on the overall quality and accuracy of the final product. Different shapes can cause varying levels of stress on the 3D printer and can result in differences in print time, material usage, and surface finish. Complex shapes with fine details and overhangs can be

more challenging to print and may require additional support structures, resulting in longer print times and increased material usage. Conversely, simple shapes with flat surfaces can be printed more quickly and with less material but may not have the same level of detail as more complex shapes. Ultimately, the shape of the object being printed will play a key role in determining the success of the 3D printing process.

The experimental results show that the proposed system can effectively monitor the quality of 3D printing, regardless of the complexity of the printed object's shape. This is evident as all the features of each printed object were detected within the acceptable tolerance range (5%) as established through the analysis of the six test objects, which included regular and irregular shapes. This suggests that the system can accurately detect any deviations in the printing process, even for objects with intricate shapes, and ensure that the final product meets the desired quality standards. This highlights the robustness and versatility of the proposed system, making it an effective tool for quality control and process optimization in the 3D printing industry.

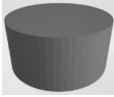
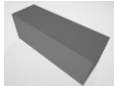

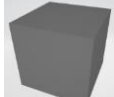



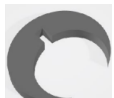
Object	Difference in dimension (Length)	Difference in dimension (Width)	Shape Correlation coefficient	Difference in infill density
	3.12%	4.63%	0.9990	2.22%
	2.5%	4.12%	0.9990	2.05%
	0.04%	0.25%	0.9988	2.90%
	0.89%	0.05%	0.9813	2.09%
	2.4%	4.6%	0.8813	3.04%
	1.2%	1.7%	0.9535	4.04%
	1.8%	4%	0.9358	1.48%
	3.35%	2.91%	0.9325	1.72%
Allowable tolerance	$\leq \pm 5\%$		>0.88	$\leq \pm 5\%$

Table 5.1: Summary of inherent displacements of 3D printing

5.2 Test on different layer

To further validate the accuracy of the proposed system, additional tests were conducted on the printed objects by monitoring their quality at different layers. In one of the tests, the printed object was deliberately damaged on a specific layer to compare the results. This test aimed to determine the sensitivity of the system in detecting any deviation in the printing process and assess its ability to identify the exact layer where the deviation occurred. The results of these tests provide valuable insights into the effectiveness and reliability of the system and allow for any necessary improvements to be made to ensure the best possible outcome for quality control in 3D printing.

In addition, the proposed system was tested on two other objects, where the printing was deliberately made to deviate from the original shape at the end of the print stage. The results, shown in Tables 2 and 3, indicated that the measurements of similarity between the camera and simulated images were within the allowable tolerance from layers #5 to #25, except at layer #30 where the deviation occurred. The deviation caused a failure in the similarity measures for width, shape, and infill in both test objects, OA and OB. This demonstrates the ability of the proposed system to detect deviations and assess the quality of 3D printing.

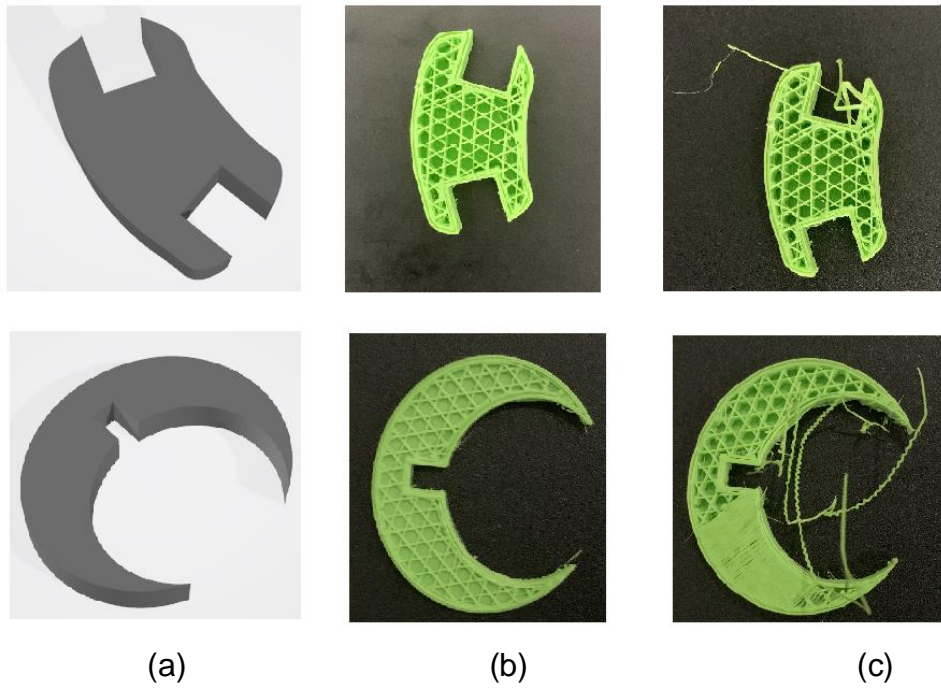


Figure 5.1: Test objects OA (upper-row) and OB (lower-row). (a) 3D shape; (b) examples of correct printing at layer #25; (c) examples of print deviation at layer #30

Layer	Difference in dimension (Length)	Difference in dimension (Width)	Shape Correlation coefficient	Difference in density
5	1.9%	2.1%	0.8996	4.33%
10	1.8%	4%	0.9358	1.48%
15	1.7%	4.8%	0.9258	3.64%
20	3.9%	4.19%	0.9063	1.02%
25	2.4%	4%	0.9544	3.66%
30 (deviated)	1.4%	30%	0.8353	7.82%

Table 5.2: Print quality assessment for test object OA.

Layer	Difference in dimension (Length)	Difference in dimension (Width)	Shape Correlation coefficient	Difference in density
5	2.34%	4.5%	0.9004	2.41%
10	3.35%	2.91%	0.9325	1.72%
15	2.39%	4.5%	0.9492	3.43%
20	2.76%	3.99%	0.8964	3.81%
25	3.03%	2.75%	0.9153	3.08%
30 (deviated)	2.98%	2.91%	0.6543	19.41%

Table 5.3: Print quality assessment for test object OB

5.3 Entire test

In order to verify the effectiveness and accuracy of the monitoring system, 35-layer printed objects were evaluated, and the process of feature extraction was demonstrated by extracting three layers from the complete test: the first layer, the middle layer, and the last layer. Finally, a graph illustrating the difference between each layer and the simulated image was presented. Figure 5.2 illustrates the results of an experiment that involved extracting size features from the first layer of a printed object. The size of the object was determined by calculating the bounding box of the extracted shape. By analysing the pixels occupied by the bounding box and using the actual unit pixel size, the size of the printed object was estimated to be 49.54 x 133.48mm.



Figure 5.2: First layer size extraction

The subsequent step involves extracting the infill density from the middle layer. Figure 5.3 depicts the printed object and the approach described in Section 3.3.2 is used to calculate the infill density of the printed layer, which is determined to be 22.36%.

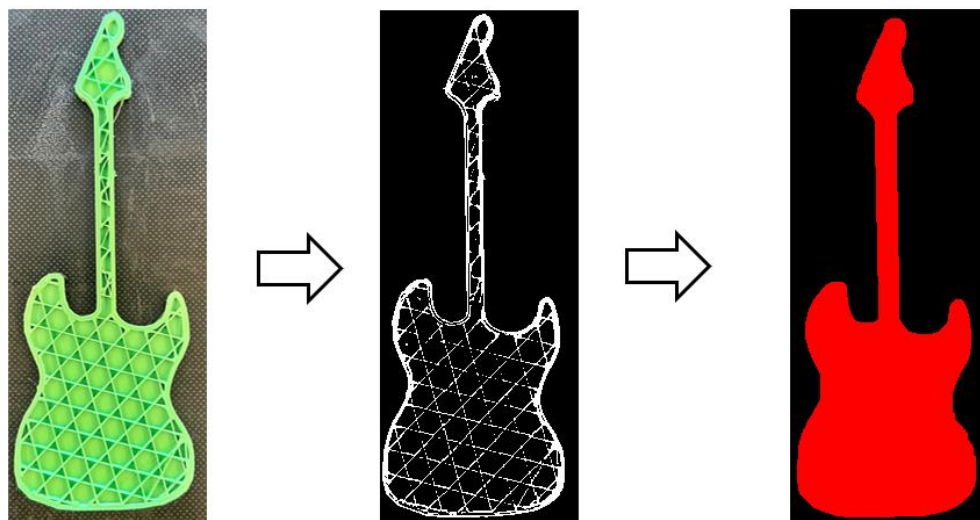


Figure 5.3: Middle layer infill density extraction

The extraction of shape from the last layer is illustrated in Figure 5.4. Following the extraction of shape features of the printing layer through image segmentation, the

shape map was transformed into an expression in the frequency domain through Fourier-Mellin transform. By performing the same operation on the simulated image, the difference in shape was evaluated by the correlation of the image.

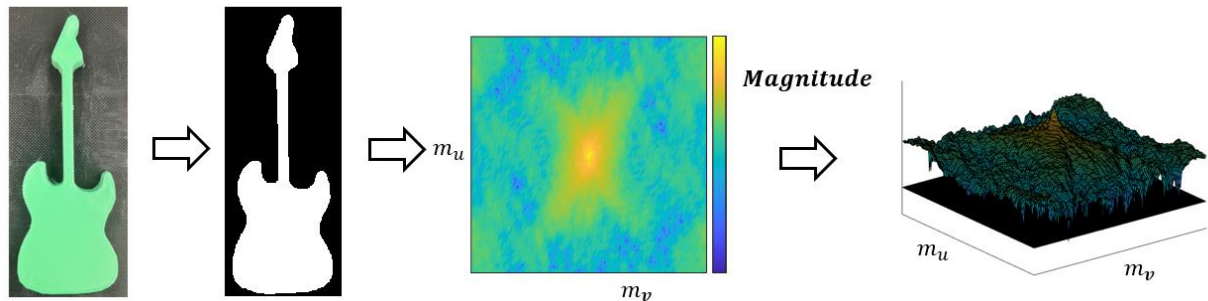


Figure 5.4: Last layer shape extraction

The results generated by the monitoring system were found to meet the expected standards. The allowable tolerance between the camera and simulated images for achieving high-quality 3D printing is within $\pm 5\%$ for the dimension and infill density. The shape correlation coefficient obtained from the related results in Figure 5.4 is also greater than 0.90, indicating a strong positive correlation between the camera and the simulated image. These results indicate that the monitoring system is capable of accurately detecting errors and deviations during the printing process and ensuring the quality of the final printed object.

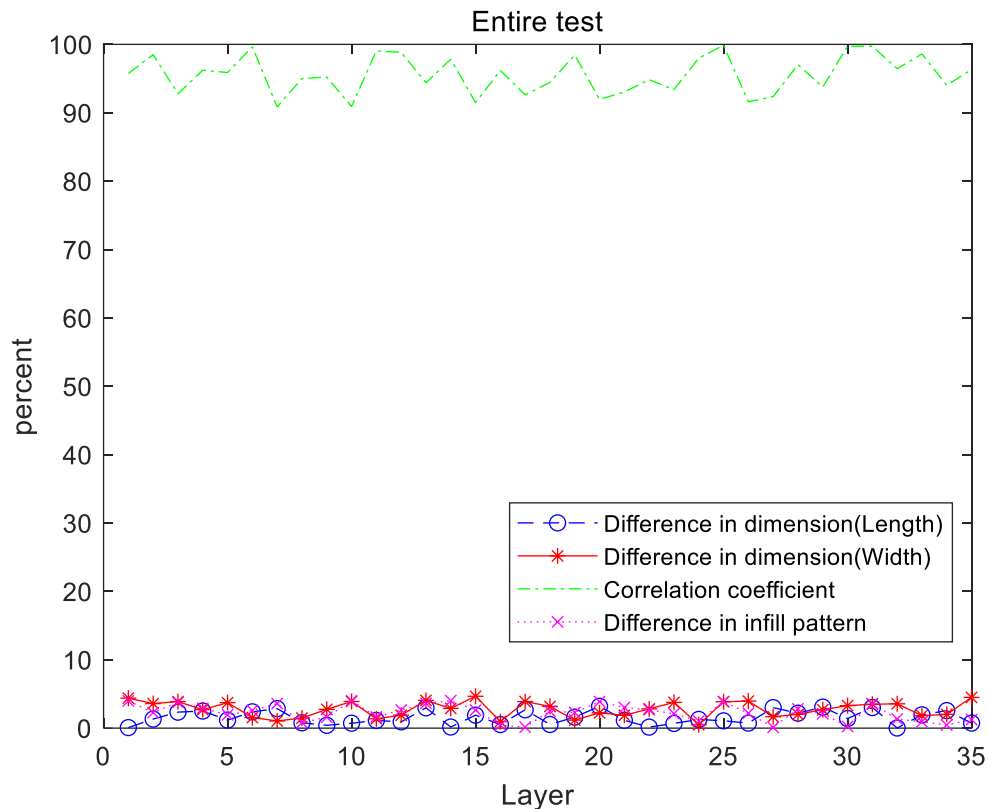


Figure 5.4: Plot of the feature evaluation parameters as the number of printed layers. A high Correlation Coefficient is indicative of a positive outcome, while low values associated with other factors are also considered favourable results.

5.4 Summary

The proposed system has been tested and validated through extensive experiments, which showed its capability to monitor the quality of 3D printing. The system was tested on six objects with different shapes, and the results showed that the features of each printed object were within the allowable tolerance range (5%). The tolerances were defined based on the extreme values of the displacements between the camera image and the simulated image created via G-code.

In addition to monitoring the printing process, the system was also tested on two objects where the printing was deliberately made to deviate from the original shape at the end of the print stage. The results indicated that the measurements of similarity between the camera and simulated images were within the allowable tolerance from layers #5 to #25, except for layer #30 where the deviation occurred. The deviation

caused a failure in the similarity measures for width, shape, and infill in both test objects. This demonstrates the ability of the proposed system to detect deviations and assess the quality of 3D printing.

Furthermore, the system was subjected to a full test on a printed object with 35 printed layers, and the results obtained for each layer were within the allowable range. This testing substantiates the system's fault tolerance and sturdiness, establishing it as a tool for enhancing process control and ensuring quality standards in the 3D printing sector.

6. Conclusions and Future Work

This project develops a computer vision-based monitoring system for quality assessment of 3D printing. The system is able to produce results which can be used to determine whether to manually stop the print when a printing error is detected to save time and materials. This chapter summarizes the research work and discusses possible future research directions. This study has the following five main contributions:

1. Emulating the process of 3D printing: A tool was developed in this project to simulate the process of 3D printing. After inputting the G-code of the printed object and the printing parameters, the image of each layer of the printed object can be generated by this tool.
2. Evaluation of size: The system employs background subtraction technology to isolate the printed object from the background, enabling accurate size determination. Analysing the length represented by each pixel and conducting blob analysis enables the estimation of the printed object's size. This estimated size is then compared to the actual size of the object, facilitating an evaluation of its size accuracy.
3. Evaluation of shape: The shape of printed objects are assessed using the Fourier-Mellin Transform (FMT) method. By mitigating the effects of rotation, scaling, and translation, this method ensures precise and consistent shape analysis. Consequently, any distortions or misalignments in the printed object, indicative of potential printing errors, can be detected.
4. Evaluation of infill density: The system calculates the infill density of printed objects by utilizing foreground extraction and image thresholding techniques. These methods combine information from both camera images and simulated images generated by simulation tools. By employing these approaches, the system achieves a precise evaluation of the fill density—a crucial quality parameter in 3D printing.

5. Evaluate print quality: In this study, three different measures have been presented for quality evaluation of 3D printing, with overall dimensions and infill ratio derived from the spatial domain, and shape correlation derived from the frequency domain. For quality 3D printing, the geometrical displacement has been shown to be within 5% for the former, and >0.88 for the latter. Also shown are some examples of their complementary nature and mutual reinforcement to illustrate the potential of their combination to extend the monitoring capability in terms of defect coverage and detection robustness.

By comparing the printed object based on the simulated and camera images using computer vision and image processing techniques, such as background subtraction and FMT, the thesis has presented a vision-based monitoring system for quality assessment of 3D printing. The method extracts a variety of geometrical features including the dimension, contour and infill pattern of the printed object, which can be used for the evaluation of printing quality to identify the defective printed parts and alter the users for time and cost-saving.

In the future work, the project will be improved from three aspects. The first one will focus on a comprehensive test of the proposed system by introducing more objects and more printing errors under different print settings to demonstrate the efficacy of the three proposed features for quality monitoring and assessment of 3D printing. The second one will detect the printing process in real time by boosting the hardware configuration. The process of 3D printing can be simulated in real time in the simulation tool by inputting printing parameters. The last one is to develop a monitoring system that can provide solutions when printing errors are detected. For example, to the extent feasible, when a printed object is detected to be partially faulty, its faulty portion can be removed during printing and reconstructed on it.

References

- All3DP. (2019). Spaghetti Detective OctoPrint Guide. All3DP. Retrieved from <https://all3dp.com/2/spaghetti-detective-octoprint-guide/>
- Alsoufi, M. S., & Elsayed, A. E. (2017). Warping deformation of desktop 3D printed parts manufactured by open source fused deposition modeling (FDM) system. *Int. J. Mech. Mechatron. Eng*, 17(11).
- Anitha, R., Arunachalam, S., & Radhakrishnan, P. (2001). Critical parameters influencing the quality of prototypes in fused deposition modelling. *Journal of Materials Processing Technology*, 118(1-3), 385-388.
- Baumann, F. W., Schuermann, M., Odefey, U., & Pfeil, M. (2017, December). From gcode to stl: Reconstruct models from 3d printing as a service. In *IOP Conference Series: Materials Science and Engineering* (Vol. 280, No. 1, p. 012033). IOP Publishing.
- Bochmann, L., Bayley, C., Helu, M., Transchel, R., Wegener, K., & Dornfeld, D. (2015). Understanding error generation in fused deposition modeling. *Surface Topography: Metrology and Properties*, 3(1), 014002.
- Bradski, G., & Kaehler, A. (2008). *Learning OpenCV: Computer vision with the OpenCV library*. " O'Reilly Media, Inc."
- Busch, S. F., Weidenbach, M., Fey, M., Schäfer, F., Probst, T., & Koch, M. (2014). Optical properties of 3D printable plastics in the THz regime and their application for 3D printed THz optics. *Journal of Infrared, Millimeter, and Terahertz Waves*, 35, 993-997.
- Ceruti, A., Liverani, A., & Bombardi, T. (2017). Augmented vision and interactive monitoring in 3D printing process. *International Journal on Interactive Design and Manufacturing (IJIDeM)*, 11, 385-395.

Cheng, Y., & Jafari, M. A. (2008). Vision-based online process control in manufacturing applications. *IEEE Transactions on Automation Science and Engineering*, 5(1), 140-153.

Dancel, R. (2019). Case study paper on additive manufacturing (3D printing technology). *Philosophy of Technology*, April.

Dey, A., Roan Eagle, I. N., & Yodo, N. (2021). A review on filament materials for fused filament fabrication. *Journal of manufacturing and materials processing*, 5(3), 69.

Dirksen, J. (2013). *Learning Three.js: the JavaScript 3D library for WebGL*. Packt Publishing Ltd.

Faes, M., Abbeloos, W., Vogeler, F., Valkenaers, H., Coppens, K., Goedemé, T., & Ferraris, E. (2016). Process monitoring of extrusion-based 3D printing via laser scanning. *arXiv preprint arXiv:1612.02219*.

Fang, E. H. C., & Kumar, S. (2019). The trends and challenges of 3D printing. *Advanced Methodologies and Technologies in Engineering and Environmental Science*, 415-423.

Fastowicz, J., Grudziński, M., Teclaw, M., & Okarma, K. (2019). Objective 3D printed surface quality assessment based on entropy of depth maps. *Entropy*, 21(1), 97.

Goodman, J. W. (2005). *Introduction to Fourier optics*. Roberts and Company publishers.

Haleem, A., & Javaid, M. (2019). Additive manufacturing applications in industry 4.0: a review. *Journal of Industrial Integration and Management*, 4(04), 1930001.

Ho, H. T., & Goecke, R. (2008, June). Optical flow estimation using fourier mellin
Development of a Vision-based Monitoring System for Quality Assessment

transform. In 2008 IEEE Conference on Computer Vision and Pattern Recognition (pp. 1-8). IEEE.

Hu, J. (2017). Study on STL-based slicing process for 3D printing. In 2017 International solid freeform fabrication symposium. University of Texas at Austin.

Jin, Y. A., He, Y., Fu, J. Z., Gan, W. F., & Lin, Z. W. (2014). Optimization of tool-path generation for material extrusion-based additive manufacturing technology. *Additive manufacturing*, 1, 32-47.

Kazik, T., & Göktoğan, A. H. (2011, April). Visual odometry based on the Fourier-Mellin transform for a rover using a monocular ground-facing camera. In 2011 IEEE International Conference on Mechatronics (pp. 469-474). IEEE.

Kerry Stevenson. (2018). Six good 3D printing troubleshooting guides. <https://www.fabbaloo.com/2018/08/six-good-3d-printing-troubleshooting-guides>.

Latif, K., Adam, A., Yusof, Y., & Kadir, A. Z. A. (2021). A review of G code, STEP, STEP-NC, and open architecture control technologies based embedded CNC systems. *The International Journal of Advanced Manufacturing Technology*, 114, 2549-2566.

Lyngby, R. A., Wilm, J., Eiríksson, E. R., Nielsen, J. B., Jensen, J. N., Aanæs, H., & Pedersen, D. B. (2017, October). In-line 3D print failure detection using computer vision. In Joint Special Interest Group meeting between euspen and ASPE: Dimensional Accuracy and Surface Finish in Additive Manufacturing.

Medelli n-Castillo, H. I., & Pedraza Torres, J. E. (2009, January). Rapid prototyping and manufacturing: A review of current technologies. In ASME International Mechanical Engineering Congress and Exposition (Vol. 43772, pp. 609-621).

Minetto, R., Volpato, N., Stolfi, J., Gregori, R. M., & Da Silva, M. V. (2017). An

optimal algorithm for 3D triangle mesh slicing. *Computer-Aided Design*, 92, 1-10.

Nikhil A. (2023). 3D Printing Processes: Material Extrusion (Part 2 of 8). Retrieved from <https://www.engineersgarage.com/3d-printing-processes-material-extrusion-part-2-8/>.

Nuchitprasitchai, S., Roggemann, M., & Pearce, J. M. (2017). Factors effecting real-time optical monitoring of fused filament 3D printing. *Progress in Additive Manufacturing*, 2, 133-149.

Nurhadiyatna, A., Jatmiko, W., Hardjono, B., Wibisono, A., Sina, I., & Mursanto, P. (2013, October). Background subtraction using gaussian mixture model enhanced by hole filling algorithm (gmmhf). In 2013 IEEE international conference on systems, man, and cybernetics (pp. 4006-4011). IEEE.

Opencv. (2019). Image thresholding. Retrieved November 10, 2019, from https://docs.opencv.org/master/d7/d4d/tutorial_py_thresholding.html.

Oropallo, W., & Piegl, L. A. (2016). Ten challenges in 3D printing. *Engineering with Computers*, 32, 135-148.

Osswald, T. A., Puentes, J., & Kattinger, J. (2018). Fused filament fabrication melting model. *Additive Manufacturing*, 22, 51-59.

Qi, W., Li, F., & Zhenzhong, L. (2010, May). Review on camera calibration. In 2010 Chinese control and decision conference (pp. 3354-3358). IEEE.

Sezgin, M., & Sankur, B. L. (2004). Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic imaging*, 13(1), 146-168.

Shahrubudin, N., Lee, T. C., & Ramlan, R. J. P. M. (2019). An overview on 3D printing technology: Technological, materials, and applications. *Procedia Manufacturing*, 35, 1286-1296.

Sitthi-Amorn, P., Ramos, J. E., Wangy, Y., Kwan, J., Lan, J., Wang, W., & Matusik, W. (2015). MultiFab: a machine vision assisted platform for multi-material 3D printing. *Acm Transactions on Graphics (Tog)*, 34(4), 1-11.

Soete, J., Badoux, B., Swolfs, Y., & Gorbatikh, L. (2019). Defect detection in 3D printed carbon fibre composites using X-ray Computed Tomography. https://www.ndt.net/article/ctc2019/papers/iCT2019_Full_paper_62.pdf, 1-8.

Straub, J. (2015). Initial work on the characterization of additive manufacturing (3D printing) using software image analysis. *Machines*, 3(2), 55-71.

Zeltmann, S. E., Gupta, N., Tsoutsos, N. G., Maniatakos, M., Rajendran, J., & Karri, R. (2016). Manufacturing and security challenges in 3D printing. *Jom*, 68(7), 1872-1881.

Zhang, Z. (2000). A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence*, 22(11), 1330-1334.

Appendix

Appendix A: Published paper on A vision-based Monitoring System for Quality Assessment of Fused Filament Fabrication (FFF) 3D Printing

A Vision-based Monitoring System for Quality Assessment of Fused Filament Fabrication (FFF) 3D Printing

JINGDONG LI

Applied Digital Signal and Image Processing Research Centre (ADSIP), University of Central Lancashire, Preston, UK

WEI QUAN

Applied Digital Signal and Image Processing Research Centre (ADSIP), University of Central Lancashire, Preston, UK

LIK-KWAN SHARK

Applied Digital Signal and Image Processing Research Centre (ADSIP), University of Central Lancashire, Preston, UK

HADLEY LAURENCE BROOKS

Advanced Digital Manufacturing Technology Research Centre (ADMT), University of Central Lancashire, Preston, UK

As one of the most popular 3D printing technology, Fused Filament Fabrication (FFF) allows intricate structures to be produced without complex manufacturing processes. However, there is a limitation of the currently available FFF 3D printers which print blindly without an ability to detect and stop upon printing deviations, incurring additional running costs due to unnecessary waste of materials and time. This has led to a novel development reported in this paper of a vision-based monitoring system for the quality assessment of 3D printing by applying advanced computer vision algorithms and imaging processing techniques. The proposed approach is through comparison between actual images of the printed layer and simulated images created by slicing CAD model via G-code generation based on the calibrated camera pose. Also presented are feature extraction methods to yield object dimension, profile and infill for quality assessment, with the system performance demonstrated based on various object geometries. Using this system makes it possible to analyze and examine the quality of 3D printing during the print process, which could identify the defective printed parts, terminate the whole process and alert the users for time and cost-savings.

CCS CONCEPTS • Computing methodologies - Modeling and simulation • Human-centered computing -Visualization

Additional Keywords and Phrases: 3D printing, modelling, simulation and quality assessment

ACM Reference Format:

Jingdong Li, Wei Quan, Lik-Kwan Shark, Hadley Laurence Brooks. 2021. A Vision-based Monitoring System for Quality Assessment of Fused Filament Fabrication (FFF) 3D Printing.

1 INTRODUCTION

Additive manufacturing widely known as 3D printing has become extremely popular in recent years. It has been used in a wide range of applications from rapid prototyping to advance manufacturing to bioprinting for biomedical engineering [1]. Fused Filament Fabrication (FFF) technology has become one of the main 3D printing approaches because of its low cost, however, the technology remains error-prone due to the quality of filament, first layer adhesion, part warping, stringing as well as improper printing parameters [2]. Because of the relatively long cycle time of operation, it usually operates unsupervised, and the defects are often identified manually after hours of printing, which results in a waste of significant amounts of time and material.

In order to monitor the printing process and assess the quality of printed objects, vision-based approaches have been widely used. Nuchitprasitchai et al. [3] proposed a single-camera system in which the camera was placed at an angle in front of the printed object. The captured images were converted to binary images to extract the shape of the

Development of a Vision-based Monitoring System for Quality Assessment

printed object. The system was able to indicate the printing error when the difference between the printed object and CAD model is greater than 5% after the shape comparison. A multiple camera system was developed by Straub et al. [4], which applied five cameras to monitor the process from different angles. By comparing the pre-recorded videos of correct printing and the current printing process, the printing failure can be detected. For each specific object, the videos of the correct printing process are required to be recorded prior to the monitoring process. Fastowicz et al. proposed a method of assessing the quality of 3D printed surfaces using depth maps [5]. It used the assumption that the quality of the printed surface was related to the entropy of depth maps. The poorer the surface quality was the bigger the entropy of depth maps presented. Sitthi-Amorn et al. [6] examined the detection of the internal imperfection of 3D printed objects by applying X-ray tomography and ultrasonic imaging, which detected embedded defects and altered printing orientation. Alessandro et al. [7] used augmented reality technology in the process of quality monitoring in which the virtual model was superimposed on the printed object and the shape difference between the model and printed object can be recognized in the different printing stages. Beyond the vision-based approaches, the electromagnetic method of terahertz non-destructive testing was used to identify internal distortions of printed objects [8]. Lowe et al. [9] proposed a method via laser scanning that used a laser as a sensor to measure the size of objects during the process. It can detect the error and provide a correction in an early stage of the process.

For the system proposed in this paper for monitoring of FFF 3D printing, it adopts the simplest hardware approach based on a single camera to offer better affordability. However, it differs from other reported single camera approaches by using not only spatial domain based direct comparison between geometrical features extracted from CAD model and camera images of printed objects, but also a frequency domain representation as an addition measure for evaluation of printing deviations. In particular, it employs the Fourier-Mellin Transform (FTM) which has been successfully applied to achieve automatic identification of different viruses in electron microscope images due to its invariant properties [10], and demonstrates its effectiveness for monitoring of 3D printing based on just one camera.

In the rest of the paper, the processing stages of the proposed vision-based monitoring system are introduced in Section 2, and it is then followed by the experimental analysis in Section 3. Finally, the concluding remarks are given in Section 4.

2 VISION-BASED MONITORING SYSTEM

Presented in this paper is a vision-based monitoring system for assessing the 3D printing quality during print processing. While the proposed system uses a single camera mounted on the top of a 3D printer to capture the images of the printed object, a developed G-Code simulator slices the 3D CAD model by using a slicing algorithm [11]. By applying the computer vision algorithms and image processing techniques, a set of features, including object dimension, profile and infill density, can be extracted from both simulated and camera images, which are used to examine the 3D printing quality and determine whether the printing process needs to be terminated. Because of the simulated images, the system is not only able to detect the defects on the surface of the printed object but also its internal imperfections. The proposed development of a vision-based monitoring system is illustrated in Figure 1.

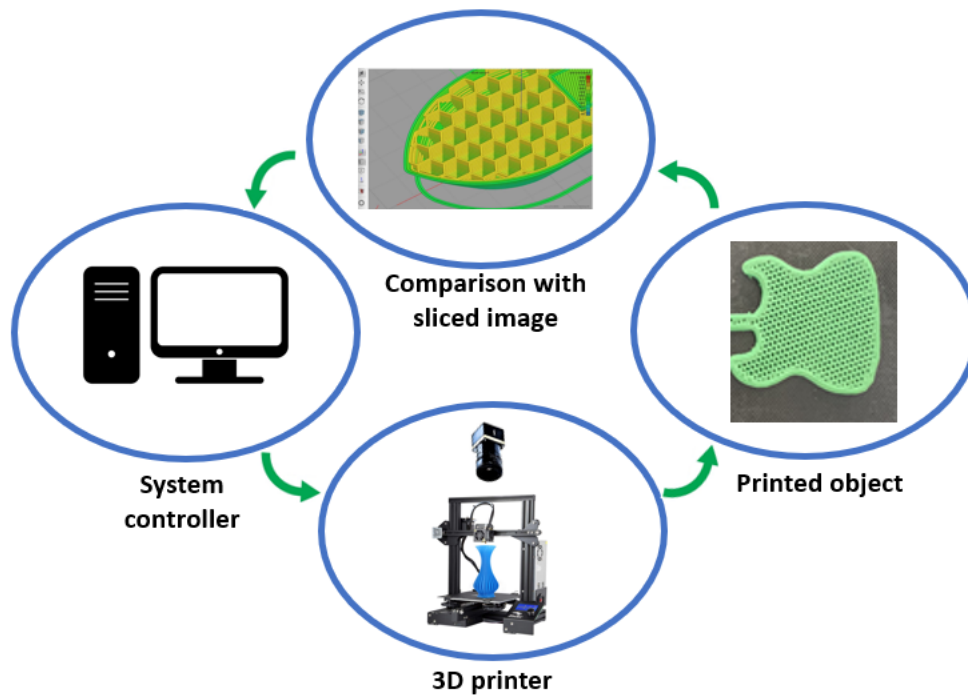


Figure 1: The proposed vision-based monitoring system for quality assessment of 3D printing

The proposed vision-based monitoring is formed by four major processing stages, including camera calibration, print simulation, feature extraction, and feature evaluation. While the camera calibration stage estimates the parameters of the camera and its pose in the world coordinate, the print simulation stage generates images for the layers of the CAD model that are to be printed. Having the camera and simulated images, the corresponding features can be extracted from both images, including the dimension, the profile and the infill density of each layer, by applying the background subtraction method and the Fourier-Mellin transform. By evaluating the differences between the corresponding features extracted from both camera and simulated images, the quality of printed objects can be assessed in order to halt printing if defects exist. The flow diagram of the processing stages is illustrated in Figure 2 and its detail is explained in the remaining part of this section.

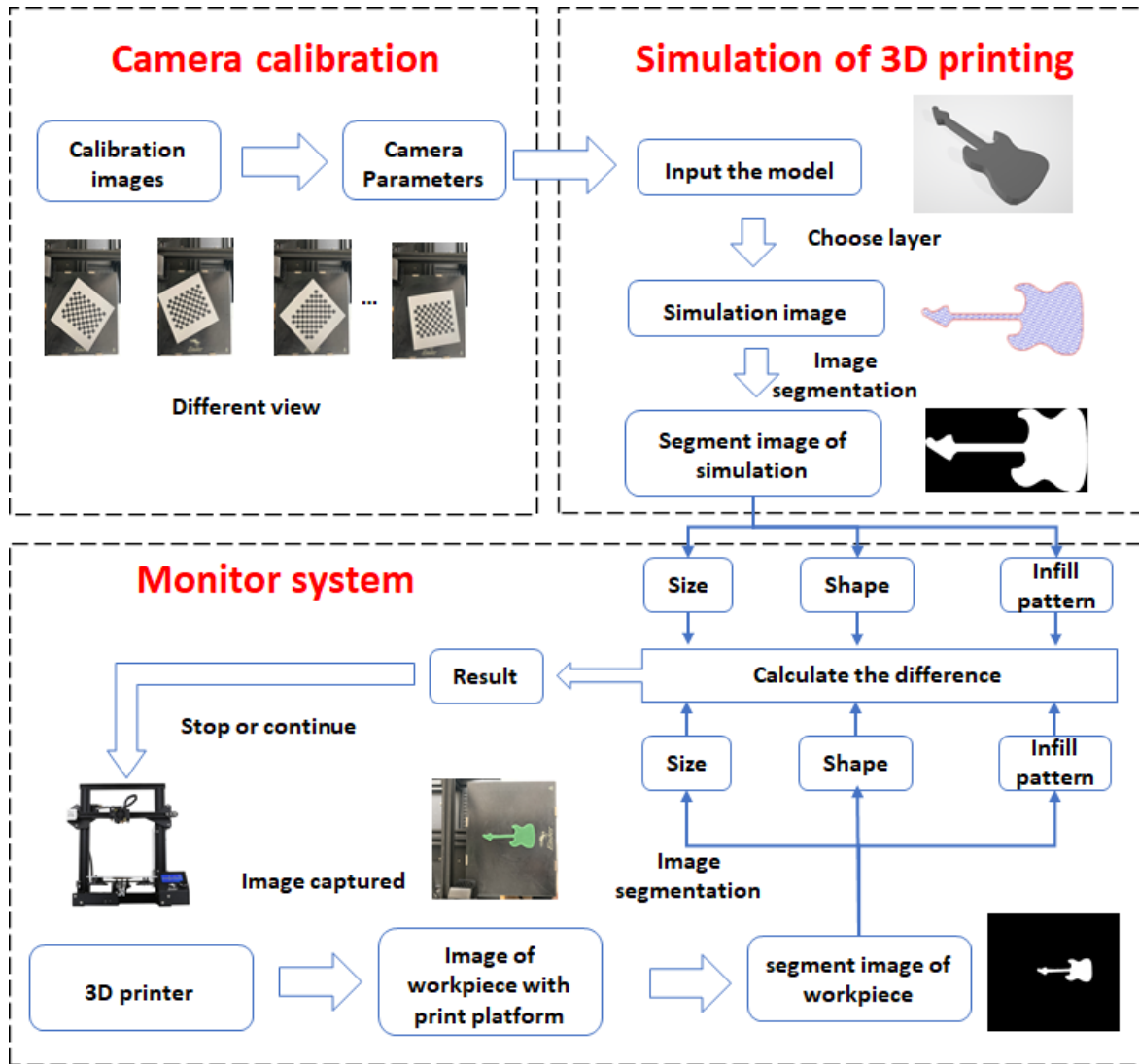


Figure 2: Flow diagram of processing stages of proposed vision-based monitoring system

2.1 Camera Calibration

The purpose of camera calibration is to estimate the camera parameters and obtain the camera position and orientation in the world coordinate system so that the correct layer images can be rendered from the CAD model. The parameters of the camera denoted by matrix P are defined as:

$$P = \begin{bmatrix} R \\ T \end{bmatrix} K \quad (1)$$

where matrix $\begin{bmatrix} R \\ T \end{bmatrix}$ contains the extrinsic parameters in terms of rotation and translation, and matrix K contains the intrinsic parameters in terms of geometric and optical characteristics of the camera. Through the camera parameters and using homogeneous coordinates, physical object points are projected to camera image points by:

$$w[x_w, y_w, 1] = [x \ y \ z] \begin{bmatrix} R \\ T \end{bmatrix} K \quad (2)$$

where w is the arbitrary scale factor, (x_w, y_w) are image points and $(x \ y \ z)$ are world coordinates of object points. Given a sufficient number of known object points (such as corner points on a checkerboard) and their corresponding image points, equation (2) can be solved to yield R , T and K . To achieve this task, the OpenCV library of camera

calibration [12] has been applied with 10 images captured from a checkerboard that has been placed above the printing platform under the camera with different orientations and the camera pose can be estimated as illustrated in Figure 3.

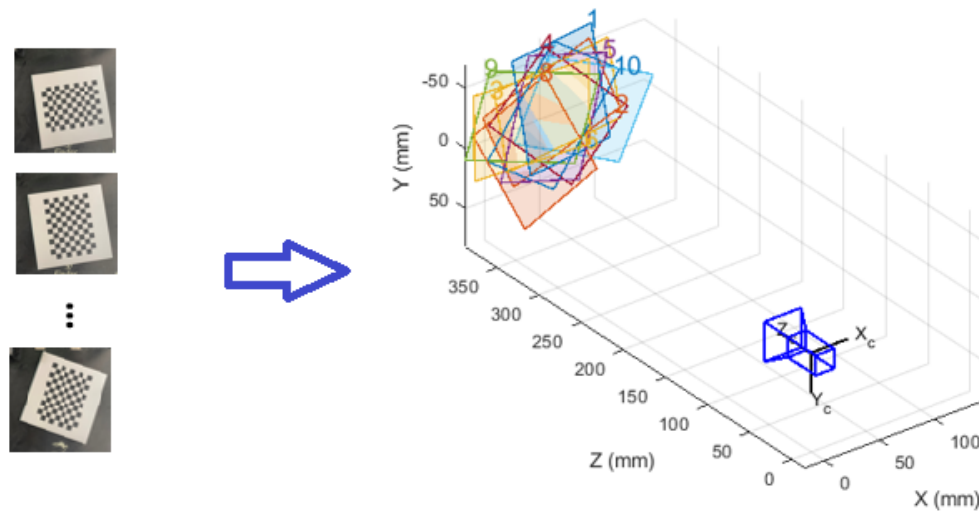


Figure 3: Camera calibration process for the recovery of camera position

2.2 Print Simulation

The print simulation is able to simulate each individual layer of a CAD model and render its image. The simulated images can then be used to compare with the actual camera images containing the corresponding printed layer in the subsequent feature extraction and evaluation stages. To be able to achieve it, a CAD model is first imported to an open-source 3D printing planning software, Slic3r, to generate G-codes for a 3D printer to execute the printing process [13]. A G-code simulator has been developed in this research in which the generated G-codes are input and executed by using every two lines of command. The commands contain the current coordinates to print and the next coordinate to move to. Each pair of coordinates are then connected by a printing line and a layer can be formed in the G-code simulator by generating the relevant printing lines. Having the estimated camera parameters and the layer, a pinhole camera model is used to render the image of the layer by applying perspective projection [14]. Despite the software, Slic3r, which can slice the CAD model and create the screenshots of the layer, it is unable to render the image with the correct intrinsic and extrinsic camera parameters. The algorithm that describes the G-code simulator is illustrated below.

LOGORITHM 1: G-code Simulator

```
// This function is using G-Code to generate expect image
function GenerateSimulationImage (G-code, Expect-Image)
//The program will read line of G-code until it's empty
while Line Is Not Empty, do
i equal to 1, i ++
command = G-code[i]
//Read current coordinate
[x[i],y[i],z[i] ]= Read Line(command)
CurrentCoordinate.x = x[i]
CurrentCoordinate.y = y[i]
CurrentCoordinate.z = z[i]
//Read next coordinate
command = G-code[i+1]
```

```

[X[i+1],Y[i+1],Z[i+1]] = Read Line(command)
NextCoordinate.X = X[i+1]
NextCoordinate.Y = Y[i+1]
NextCoordinate.Z = Z[i+1]
//draw the line based on the current coordinate and next coordinate
Exptce-Image = DrawLine(CurrentCoordinate, NextCoordinate)
end
end

```

Having obtained the camera images and corresponding layer images, the features can be extracted from both images to assess the quality of printing. Three types of features have been used for assessing the quality of 3D printing, namely (a) the dimension of the printed layer, (b) the profile of the printed layer, and (c) the infill density. In order to obtain the features, two sub-processing steps are designed and implemented in this research, namely, (a) background subtraction, and (b) Fourier-Mellin transform (FMT). While the background subtraction separates the printed layers and background in which the dimension of the printer layers can be measured, FMT decouples the variations between the camera and simulated images for the shape comparison.

2.2.1 Background Subtraction

The background subtraction only needs to be applied to the camera images as the simulated images already have the printed object and background separated in two different colours when they are generated by the G-coder simulator. According to the method proposed by Benezeth et al. [15], 10 images that contain only the printer bed of the 3D printer are captured by the camera before the printing and the mean image of them is used as the background image. Since the camera condition and the illumination stay unchanged, the printed object can be separated from the camera images by subtracting the background images with the help of morphological operation [16]. The result of background subtraction applied on camera and simulated images are shown in Figure 4, respectively.

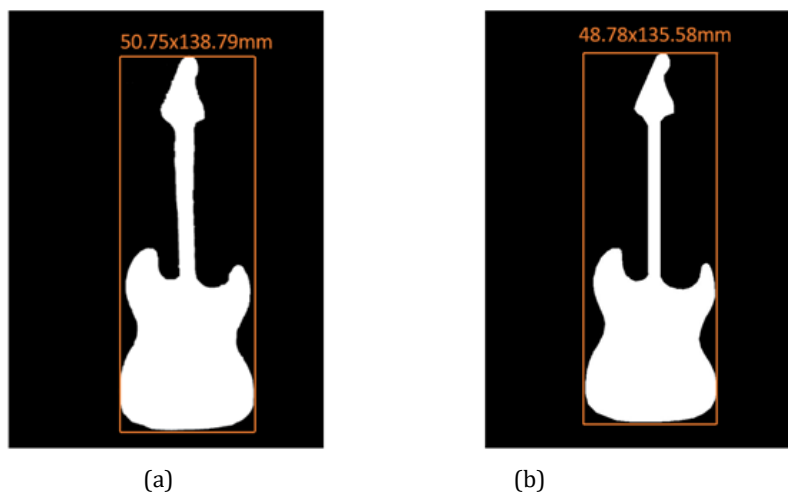


Figure 4: Examples of background subtraction: (a) from the camera image; (b) from the simulated image

2.2.2 Fourier-Mellin Transform

Fourier-Mellin transform (FMT) is based on Fourier transform analysis, which converts the images to erase the effects of rotation, translation and scaling [17]. By using polar coordinates to represent an image as $f(r, \theta)$, its Fourier-Mellin transform is given by:

$$M_f(u, v) = \frac{1}{2\pi} \int_0^\infty \int_0^{2\pi} f(r, \theta) r^{-ju} e^{-jv\theta} d\theta \frac{dr}{r} \quad (3)$$

where the Fourier integral is seen to be combined with the Mellin integral with the former applied along the angular direction and the latter along the radial direction. If two images have a rotation and scaling difference, such that $f_1(r, \theta) = f_2(\alpha r, \theta + \beta)$, then their FMTs are related as follows:

$$M_{f_1}(u, v) = \alpha^{-ju} e^{-jv\beta} M_{f_2}(u, v) \quad (4)$$

where the magnitudes of $M_{f_1}(u, v)$ and $M_{f_2}(u, v)$ have a translation in the r and θ axes. By substituting $r = e^\rho$, the FMT can be expressed as a Fourier transformation:

$$M_f(u, v) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_0^{2\pi} f(e^\rho, \theta) r^{-ju\rho} e^{-jv\theta} d\theta d\rho \quad (6)$$

Therefore, by applying the Fourier transform to an image and remapping the transformed spectrum to log-polar coordinates, a new representation of translation, rotation and scaling invariant is created. Figure 5 illustrates the process of FMT applied to two images, which contain the same object with the variation of translation, rotation and scaling. The relative translation of the object in both images is eliminated after the first Fourier transform, whereas the relative rotation and scaling are conveyed as the translation after remapping the transformed spectra to log-polar coordinates. The translation in the log-polar domain is then decoupled by applying the second Fourier transform. It can be seen that the resulting images are almost identical without the variation of translation, rotation and scaling of the same object. In this research, FMT is not only used to extract the feature for comparing the profile of printed layers but also to extract the feature for evaluating the infill pattern of the printed layers.

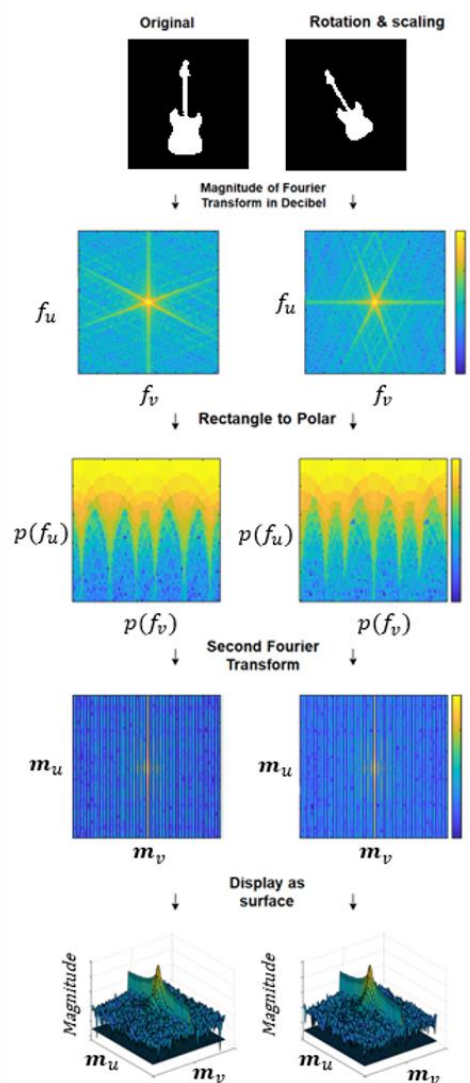


Figure 5: An illustration of Fourier-Mellin transform

2.3 Feature Evaluation

Having gone through the background subtraction and FMT, three proposed features can be measured on the camera and simulated images. For the dimension of the printed layer, it is obtained through the bounding box as shown in Figure 4, in which the length and width of the box are described as the dimension of the printed layer. The pixel-to-mm conversion of the camera image is obtained in the camera calibration process by knowing the size of the checkerboard. The dimension of the CAD model is found by using the G-code simulator.

For the profile comparison, FMT is applied to the printed layer acquired from the background subtraction. Since the variation of translation, rotation and scaling are decoupled, the contour of the printed layer in both camera and simulate images can be compared by calculating the 2D correlation coefficient between their resulting FMT images, as shown in Figure 7. In the figure, it can be seen that the FMT images of the camera and simulate images are almost identical for the printed layer of a guitar and their correlation coefficient is 0.9931.

The infill pattern of the printed layer can also be examined by calculating the infill intensity in which the number of pixels representing the infill-pattern is counted against the total number of pixels inside its contour. Figure 7 demonstrates an example of the intensity calculation in which the pixels in white represent infill and the pixels in red represent the total area, giving an infill intensity of 20.57% for the printed layer.

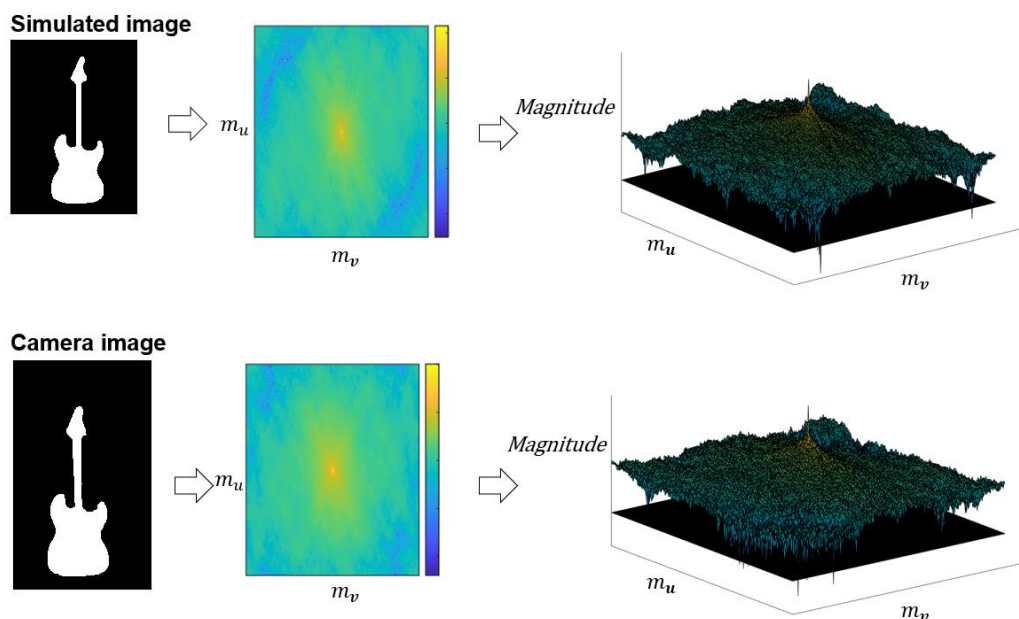


Figure 6: Comparison of FMT images for the profile of printed layer

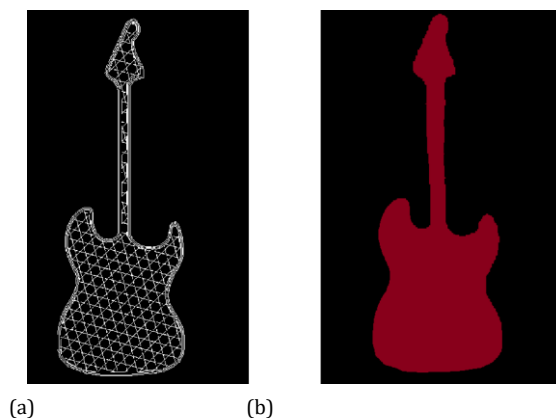


Figure 7: Infill intensity calculation: (a) pixels of infill in white; (b) overall pixels in red

3 EXPERIMENTAL ANALYSIS

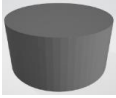
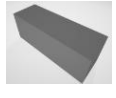

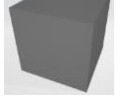



Due to the imperfection of the camera calibration and feature extraction process, there are inherent displacements between the features extracted from the camera image of the printed layer and the corresponding simulated image created via G-code, which need to be tolerated and excluded from the detection of printing errors in the quality assessment. Using a Creality Ender 3 3D printer, seven objects were printed to investigate the range of inherent displacement and define tolerance for quality monitoring and process control, which included regular and irregular shapes as shown in Table 1. Based on the extreme values of the displacements observed from Table 1 for the three proposed features, the allowable tolerance between the camera and simulated images for quality 3D printing are seen to be within $\pm 5\%$ for the dimension and infill pattern with >0.88 for the correlation coefficient.

The proposed system was also tested for monitoring 3D printing of two other objects shown in Figure 9, in which the printing of both objects was deliberately made to deviate from the original shape at the end of the print stage to see if the quality variation can be detected by the proposed method. Tables 2 and 3 illustrate the print quality assessment for test objects OA and OB, respectively. The measurements of the three features were undertaken in every 5 layers. From both tables, it is seen that the measurements of similarity between the camera and simulated images are always

Development of a Vision-based Monitoring System for Quality Assessment

within the allowable tolerance from layers #5 to #25 (i.e., good quality of printing) except layer #30 where the print deviation happened. While the print deviation of test object OA failed the similarity measures in the width, profile and infill of the object, a similar phenomenon is observed for test object OB in which the print deviation causes the failure of similarity measure in the profile and infill of the object.

Table 1: Summary of inherent displacements of 3D printing

Object	Difference in dimension (Length)	Difference in dimension (Width)	Correlation coefficient	Difference in infill pattern
	3.12%	4.63%	0.9990	2.22%
	2.5%	4.12%	0.9990	2.05%
	0.04%	0.25%	0.9988	2.90%
	0.89%	0.05%	0.9813	2.09%
	2.4%	4.6%	0.8813	3.04%
	1.0%	2.4%	0.9931	1.89%
	1.2%	1.7%	0.9535	4.04%
Allowable tolerance	<±5%		>0.88	<±5%

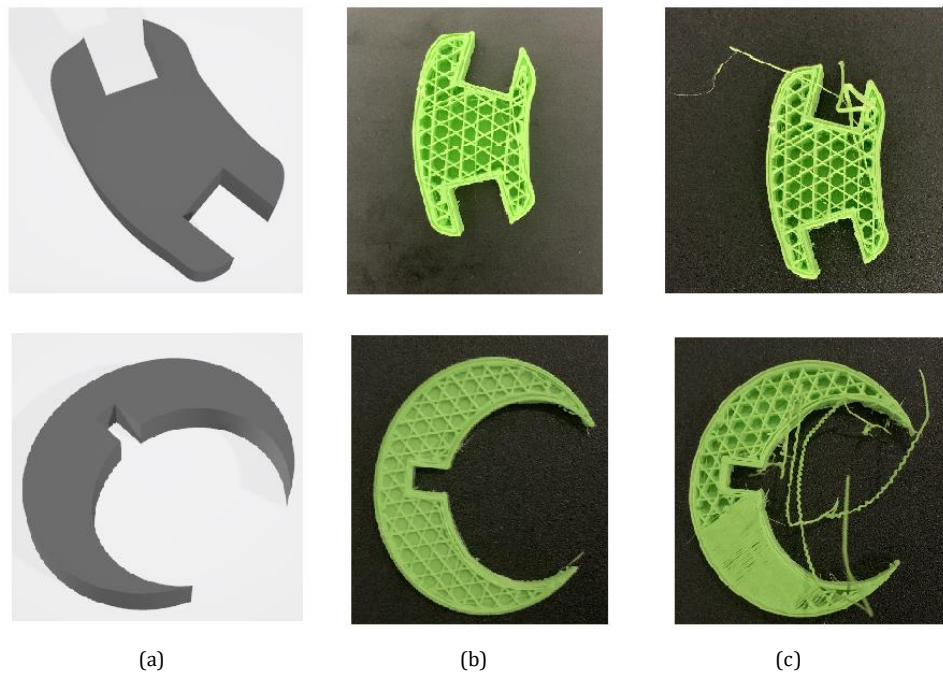


Figure 8: Test objects OA (upper-row) and OB (lower-row): (a) 3D shape; (b) examples of correct printing at layer #25; (c) examples of print deviation at layer #30

Table 2: Print quality assessment for test object OA.

Layer	Difference in dimension (Length)	Difference in dimension (Width)	Correlation coefficient	Difference in infill pattern
5	1.9%	2.1%	0.8996	4.33%
10	1.8%	4%	0.9358	1.48%
15	1.7%	4.8%	0.9258	3.64%
20	3.9%	4.19%	0.9063	1.02%
25	2.4%	4%	0.9544	3.66%
30 (deviated)	1.4%	30%	0.8353	7.82%

Table 3: Print quality assessment for test object OB

Layer	Difference in dimension (Length)	Difference in dimension (Width)	Correlation coefficient	Difference in intensity
5	2.34%	4.5%	0.9004	2.41%
10	3.35%	2.91%	0.9325	1.72%
15	2.39%	4.5%	0.9492	3.43%
20	2.76%	3.99%	0.8964	3.81%
25	3.03%	2.75%	0.9153	3.08%
30 (deviated)	2.98%	2.91%	0.6543	19.41%

4 CONCLUSIONS

In this study, three different measures have been presented for quality evaluation of 3D printing, with overall dimensions and infill ratio derived from the spatial domain, and profile correlation derived from the frequency domain. For quality 3D printing, the geometrical displacement has been shown to be within 5% for the former, and >0.88 for the latter. Also shown are some examples of their complementary nature and mutual reinforcement to illustrate the potential of their combination to extend the monitoring capability in terms of defect coverage and detection robustness.

By comparing the printed object based on the simulated and camera images using computer vision and image processing techniques, such as background subtraction and FMT, the paper has presented a vision-based monitoring system for quality assessment of 3D printing. The method extracts a variety of geometrical features including the dimension, contour and infill pattern of the printed object, which can be used for the evaluation of printing quality to identify the defective printed parts and alter the users for time and cost-saving. Follow-on work will focus on a comprehensive test of the proposed system by introducing more objects and more printing errors under different print settings to demonstrate the efficacy of the three proposed features for quality monitoring and assessment of 3D printing.

REFERENCES

- [1] Baumann, F. and Roller, D., 2016. Vision based error detection for 3D printing processes. MATEC Web of Conferences, 59, p.06003.
- [2] Bochmann, L., Bayley, C., Helu, M., Transchel, R., Wegener, K. and Dornfeld, D., 2015. Understanding error generation in fused deposition modeling. Surface Topography: Metrology and Properties, 3(1), p.014002.
- [3] Nuchitprasitchai, S., Roggemann, M. and Pearce, J., 2017. Factors effecting real-time optical monitoring of fused filament 3D printing. Progress in Additive Manufacturing, 2(3), pp.133-149.
- [4] Straub, J. (2015). Initial work on the characterization of additive manufacturing (3D printing) using software image analysis. Machines, 3(2), 55-71.
- [5] Fastowicz, J., Grudziński, M., Teclaw, M., & Okarma, K. (2019). Objective 3D printed surface quality assessment based on entropy of depth maps. Entropy, 21(1), 97.
- [6] Sitthi-Amorn, P., Ramos, J. E., Wangy, Y., Kwan, J., Lan, J., Wang, W., & Matusik, W. (2015). MultiFab: a machine vision assisted platform for multi-material 3D printing. Acm Transactions on Graphics (Tog), 34(4), 1-11.
- [7] Ceruti, A., Liverani, A., & Bombardi, T. (2017). Augmented vision and interactive monitoring in 3D printing process. International Journal on Interactive Design and Manufacturing (IJIDeM), 11(2), 385-395.
- [8] Zeltmann, S. E., Gupta, N., Tsoutsos, N. G., Maniatakos, M., Rajendran, J., & Karri, R. (2016). Manufacturing and security challenges in 3D printing. Jom, 68(7), 1872-1881.
- [9] Faes, M., Abbeloos, W., Vogeler, F., Valkenaers, H., Coppens, K., Goedemé, T., & Ferraris, E. (2016). Process monitoring of extrusion based 3D printing via laser scanning. arXiv preprint arXiv:1612.02219.
- [10] Shark L.-K., Matuszewski B. & Hall G. (1998). New template matching approach for automatic classification of viruses in electron microscope images. In Proceedings of Medical Image Understanding and Analysis, Leeds, UK, pp157-160.
- [11] Lyngby, R. A., Wilm, J., Eiríksson, E. R., Nielsen, J. B., Jensen, J. N., Aanæs, H., & Pedersen, D. B. (2017). In-line 3D print failure detection using computer vision. In Joint Special Interest Group meeting between euspen and ASPE: Dimensional Accuracy and Surface Finish in Additive Manufacturing.

- [12] Bradski, G. (2000). The openCV library. *Dr. Dobb's Journal: Software Tools for the Professional Programmer*, 25(11), 120-123.
- [13] Zeltmann, S. E., Gupta, N., Tsoutsos, N. G., Maniatakos, M., Rajendran, J., & Karri, R. (2016). Manufacturing and security challenges in 3D printing. *Jom*, 68(7), 1872-1881
- [14] Milnthorpe, G., McCormick, M. and Davies, N., 2002. Computer modeling of lens arrays for integral image rendering. *Proceedings 20th Eurographics UK Conference*,
- [15] Benezeth, Y., Jodoin, P. M., Emile, B., Laurent, H., & Rosenberger, C. (2008, December). Review and evaluation of commonly-implemented background subtraction algorithms. In *2008 19th International Conference on Pattern Recognition* (pp. 1-4).
- [16] Zhang Z, 2000, "A Flexible New Technique for Camera Calibration", *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22(11) 1330-1334.
- [17] Sthoduka.github.io. 2021. Fourier-Mellin transform. [online] Available at: <https://sthoduka.github.io/imreg_fmt/docs/fourier-mellin-transform/> [Accessed 30 November 2021].