University of Nottingham
UK | CHINA | MALAYSIA

Improving the Precision of 3D Printing in Vat Photopolymerization Using Machine Learning-Driven Boundary Prediction

By Yeting Ma

Supervisors Dr. Yinfeng He
Prof. Haonan Li
Dr. Gongyu Liu

A master thesis submitted for the degree of Master of Mechanical Engineering of Nottingham University

Department of Mechanical, Materials, andManufacturing Engineering

May 2024

(This page is left blank)

Abstract

Recent advancements in Digital Light Processing (DLP) 3D printing technology have led to improved precision and speed, along with the development of specialized resin materials catering to diverse industry needs. Like other pixel-based additive manufacturing techniques, challenges persist in achieving DLP printing fidelity due to hardware limitations, particularly in reproducing structures smaller than one pixel. The integration of grayscale pixels has been a latest strategy to address this issue, but optimal allocation remains a challenge.

Photopolymerization-based 3D printing methods often involve direct or predictive modeling of the cure process to anticipate ink solidification under varying UV intensities, typically utilizing inverse methods to determine the required light distribution. While effective, these models typically require measurement of ink cure degree or mechanical performance, necessitating additional time-consuming and intricate characterization steps. Moreover, the computational demands of these simulations often call for high-performance computing resources, adding complexity to the process.

To simplify the allocation of grayscale pixels for optimal light distribution, the application of machine learning (ML) has emerged as a promising solution. ML offers a data-driven approach to swiftly identify optimal configurations, reducing costs and enhancing printing efficiency. Researchers have successfully employed ML techniques to automate dataset refinement, mitigate scattering effects, improve printing accuracy for large-scale features, and efficiently obtain stress-strain curves. Recent advancements have demonstrated the efficacy of neural network-based ML models in predicting projection patterns, leading to significant enhancements in vat photopolymerization 3D printing precision. While these approaches show promise, some may require substantial characterization data and computational resources that may not be universally available in all research settings.

To overcome these challenges, a data-driven approach leveraging ML techniques has been proposed in this thesis. By using ML algorithms to predict grayscale pixel allocation, this approach aims to streamline the printing workflow, optimize material usage, and enhance printing precision. Through the implementation of a chessboard patterning strategy and an automated data refining and augmentation workflow, the proposed approach demonstrates increased efficiency and effectiveness. This advancement in DLP 3D printing technology, coupled with specialized resin

materials catering to diverse industry needs, signifies a significant progression in the field. The integration of ML techniques not only enhances accuracy and efficiency in vat photopolymerization but also facilitates easy sharing and utilization of the trained model across multiple users without the need for extensive computing resources.

Acknowledgements

I would like to express my deepest gratitude to Dr. Yinfeng He for his invaluable guidance, unwavering support, and patience throughout the entire duration of this research project. Dr. He not only possesses an impressive depth of knowledge and scholarly expertise but also demonstrates a rare blend of wisdom and patience in his mentorship. His keen academic insights and rigorous approach to research have profoundly influenced my intellectual growth and instilled in me a deeper appreciation for the pursuit of knowledge. The guidance provided by my supervisor was of great significance to me throughout the process of completing this master's thesis.

I am also particularly grateful to Prof. Haonan Li for providing me with a platform to explore my research interests and for his invaluable mentorship that has shaped the trajectory of my academic journey. His mentorship has been instrumental in helping me navigate the complexities of research, and his constructive feedback has challenged me to strive for excellence in my scholarly pursuits. I am truly grateful for the opportunity to learn and grow under his tutelage. His unwavering belief in my abilities and his encouragement to push the boundaries of my research have been instrumental in shaping the outcomes of this work.

Furthermore, I would like to express my sincere gratitude to Dr. Gongyu Liu from the China Beacons Institute at the University of Nottingham Ningbo China and Dr. Yongjie Zhao from Ningbo University for their invaluable support throughout this project. Dr. Zhao's expertise and experience in machine vision have been instrumental in guiding my research. Dr. Liu's encouragement to explore new ideas and his high standards have motivated me to delve deeper into the academic realm and develop my own perspectives.

In closing, I extend my appreciation to the research staff and colleagues at the Advanced Manufacturing Centre whose collaboration and assistance have enriched the research experience and contributed to the success of this study. I am grateful to my family for their understanding and support during this challenging academic journey. Their encouragement and belief in my abilities have been a constant source of motivation.

This research would not have been completed without the support and contributions of all those mentioned above. Thank you for being part of this academic endeavor.

Publications

Paper accepted

[1] **Yeting Ma**, Zhennan Tian, Bixuan Wang, Yongjie Zhao, Yi Nie, Ricky Wildman, Hao Nan Li, Yinfeng He. Enhancing the 3D printing fidelity of vat photopolymerization with machine learning-driven boundary prediction[J]. Materials & Design, 2024: 112978.

Patents under review

[1] Gongyu Liu, Hao Nan Li, **Yeting Ma**, Yinfeng He, Zhennan Tian. A 3D printing grayscale compensation method, an electronic device, and a readable storage medium. CN116945605A. 2023.08.29

Contents

List of Figures and Table

Chapter 1 Introduction

1.1 Background

In recent years, Digital light processing (DLP) 3D printing technology has undergone significant advancements. The continuous refinement and diversification of light-curing resin materials have led to enhancements in 3D printing precision and speed. Concurrently, the printers have seen a gradual increase in size, enabling the fabrication of larger-scale objects. A variety of resin materials with specialized properties have emerged, such as high temperature resistance and chemical corrosion resistance, catering to the specific demands of various industries.

DLP 3D printing technology effectively balances printing accuracy and cost-efficiency, making it a prevalent choice across diverse sectors, including ceramic manufacturing, aerospace engine component production, automotive part prototyping, and biofabrication of tissues and organs.

While there have been notable advancements in DLP printing, challenges persist in terms of fidelity, with enhancements often reliant on hardware progress. Enhancing the fidelity of 3D-printed objects involves optimizing the shape and structure to achieve characteristics such as smoother edges. By focusing on fidelity, the aim is to ensure that the actual printed results closely match the desired design, with a high level of detail and precision. At the core of this printing technology lies the Digital Micromirror Device (DMD) chip, which governs the UV light pattern. Comprising millions of microlenses, each microlens corresponds to a pixel in the printed image. However, a limitation exists in that it cannot reproduce structures smaller than one pixel or print a half-pixel border, thereby constraining printing precision and resulting in a zig-zag outline of the printed structure.

One prevalent strategy involves the incorporation of grayscale pixels to facilitate the creation of partially solidified areas, thereby aiding in the smoothing of stepped edges [1] (see Figure 1-1 i). Nevertheless, a key challenge lies in determining the optimal allocation of grayscale pixels. Current grayscale prediction techniques in DLP 3D printing encompass direct or predictive modeling of the curing process to anticipate ink solidification under varying UV intensities. This often involves utilizing inverse methods to ascertain the necessary light distribution. However,

these models typically necessitate measurements of ink solidification or mechanical properties, necessitating the inclusion of additional time-consuming characterization steps in the workflow. Furthermore, processing extensive data sets often mandates the use of high-performance computers or parallel computing clusters, thereby amplifying the complexity of the overall process.

The incorporation of machine learning (ML) techniques signifies a notable progression in the field. By harnessing ML algorithms to forecast the allocation of grayscale pixels, it enables the streamlining of the printing workflow, optimization of material utilization, reduction of waste, and cost savings. Furthermore, the automated allocation of grayscale pixels through ML simplifies operations, enhances printing precision, diminishes stepped edges, and refines the printed output [2] (see Figure 1-1 ii). Nonetheless, the training of ML models demands a substantial volume of representative data and computational resources, presenting challenges for laboratories lacking access to such resources.

To address these challenges, this thesis presents a data-driven approach designed to enhance shape accuracy effectively, aiming to produce structures that closely match the desired geometry more effectively than existing methods. The proposed method introduces a data acquisition and model training strategy that reduces the data collection requirements while enhancing the printing quality of the manufacturing process. This methodology involves two sequential ML steps: the first step focuses on data acquisition and expansion, while the second step involves a model that uncovers the relationships between projection and printed patterns, leveraging this information to achieve improved structures. The effectiveness of this approach was validated across three different commercially available vat photopolymerization resins and various challenging structures, demonstrating significant enhancements in all scenarios, including complex 3D objects.

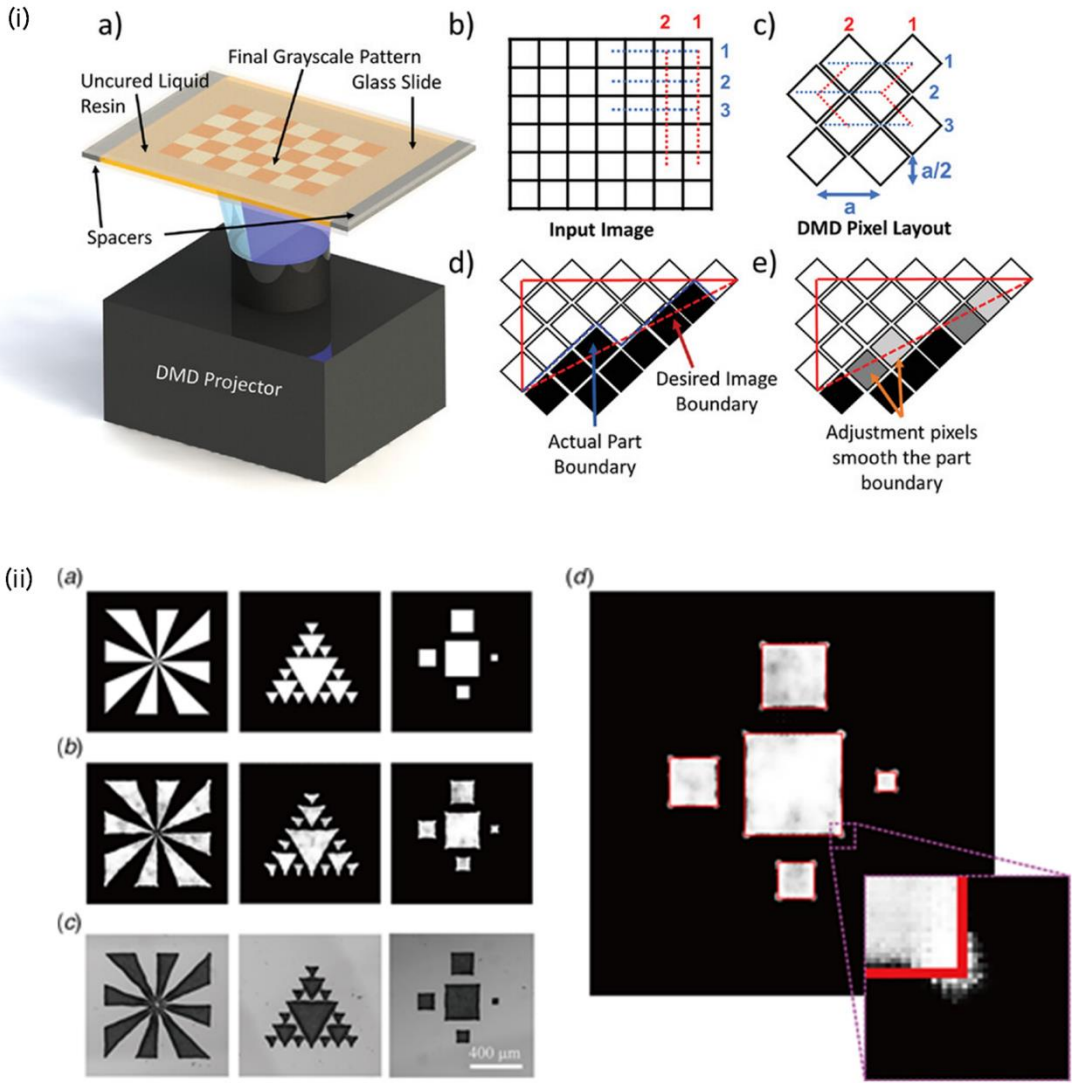


Figure 1-1 The application of machine learning in DLP 3D printing: (i) Grayscale adjustment pixels in DMD-based DLP printing can be utilized to match the smooth desired boundaries [1]. (ii) Applying machine learning methods to optimize grayscale pixels for improved precision in DLP printing [2].

1.2 Project aims and thesis structure

To enhance the precision of DLP 3D printing, this research project employed an ML model based on U-Net to predict grayscale pixels, aiding in the smoothing of printing edges. Considering the high time and computational costs associated with training ML models, innovative data collection and processing methods were introduced to reduce the model's training time significantly. Given that collecting training data for ML models typically requires a substantial amount of time and labor costs, this study utilized data augmentation techniques to enrich the training dataset, thereby reducing the reliance on experimental data.

Therefore, the objectives of this thesis are to operationalize the data-driven methodology as follows:

(i) Develop automated data acquisition and processing methods to enhance the efficiency of data acquisition.

(ii) Employ ML strategies to enhance DLP printing effects and enable the printing of sub-pixel features.

(iii) Utilize data augmentation techniques to expand model training datasets and minimize the volume of experimental data needed.

The outline of each chapter is as follows (see Figure 1-2):

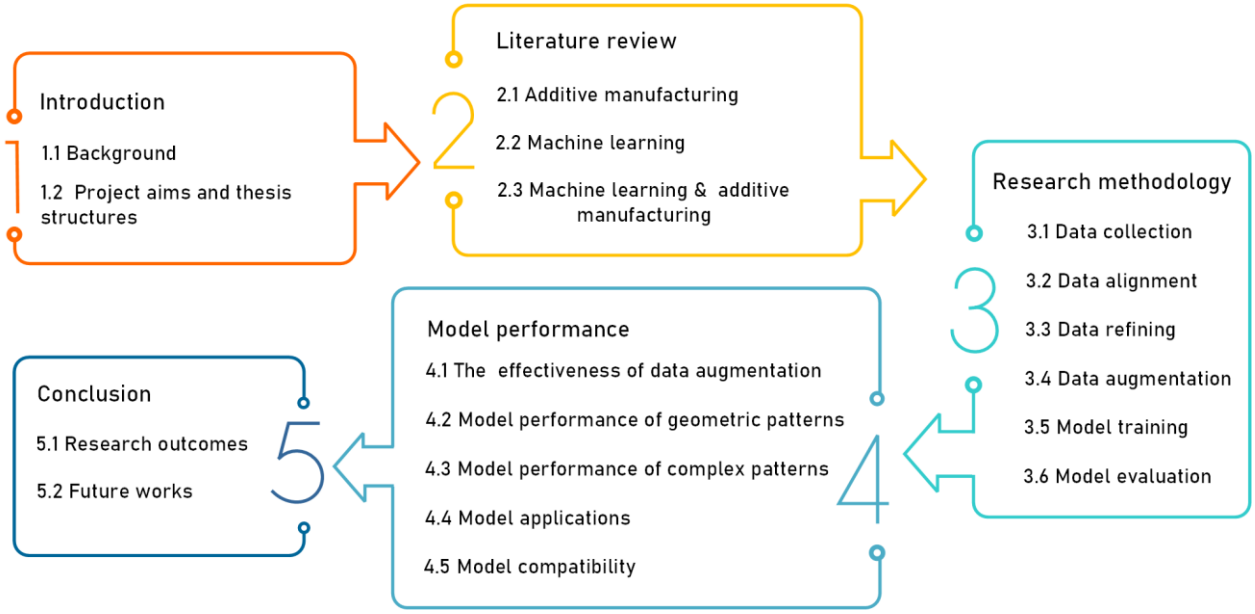


Figure 1-2 Scope of the study.

Chapter 2: Literature review introduces a detailed review on (i) the utilization of vat photopolymerization and other additive manufacturing techniques, (ii) the advancements and implementation of machine learning methodologies, and (iii) the integration of machine learning techniques within the realm of additive manufacturing applications.

Chapter 3: Research methodology introduces the specific steps to implement the data-driven method: (i) Data acquisition and processing, (ii) Data augmentation, (iii) Model training and (iv) Model evaluation.

Chapter 4: Model performance introduces the efficacy and compatibility of the trained model, and evaluates its performance in practical applications.

Chapter 5: Conclusion summaries the research outcomes and present future work for this thesis.

Chapter 2 Literature review

2.1 Additive Manufacturing

Additive Manufacturing (AM) technology, also known as 3D printing, is a manufacturing process that builds products layer by layer using materials such as powder, wire, and plate, based on a three-dimensional CAD model [3]. Unlike traditional manufacturing, it does not require specific molds or tools.

AM technology is rapidly advancing, finding applications in various industries such as aerospace [4], biomedical [5, 6], and automobile manufacturing [7] (see Figure 2-1). In aerospace, it meets demands for high precision and complex shapes, as seen in the production of complex-shaped aluminum alloy parts for the Chang'e-4 relay satellite. In biomedicine, it has been used for cranial implants, braces, and hearing aids, with 3D printed medical devices approved by the CFDA for personalized use. In automobile manufacturing, it offers new possibilities for lightweight body design and flexible production [8].

According to ISO ASTM 52900:2015, which standardizes terminology for materials testing and technology development, AM can be categorized into seven basic types: Vat photopolymerization (VP), Material extrusion, Material jetting, Binder jetting, Powder bed fusion, Direct energy deposition, and Sheet lamination. These categories use different forms of materials such as granular powders, filaments, liquids, and flake sheets and can be further divided into subcategories. Among them, VP is widely used for its high precision and low manufacturing cost.

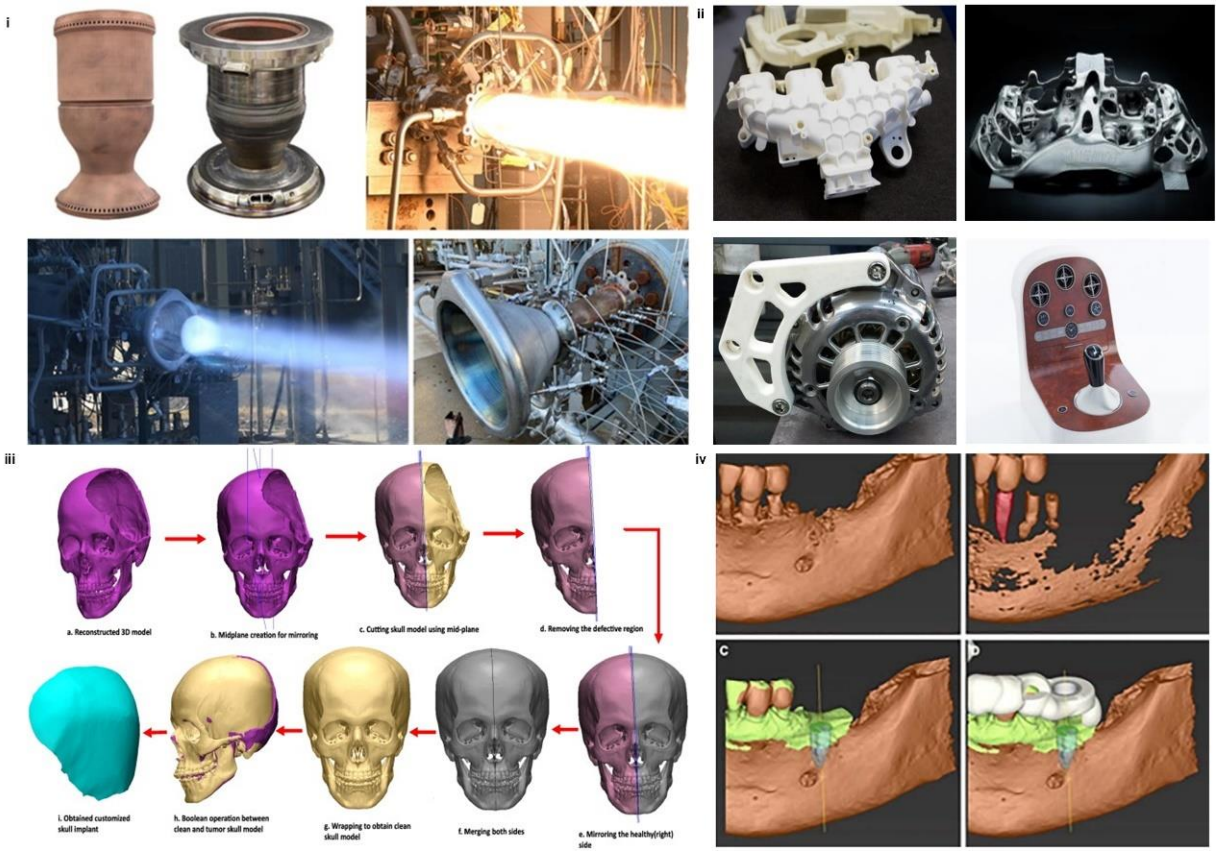


Figure 2-1 Various applications of additive manufacturing. (i) Fully AM thrust chamber assemblies hot-fire tested at NASA MSFC [4]; (ii) Several car components, including the alternator bracket, center console, and brake caliper, produced using AM technologies [7]. (iii) Steps involved in designing cranial reconstruction implants using 3D printing [6] and (iv) Design and manufacture implants through AM [5].

2.1.1 Vat Photopolymerization

VP uses UV light to harden liquid photopolymer in specific areas of a tank. Photopolymers, essentially thermosetting polymers, cross-link when exposed to UV or visible light through photopolymerization, resulting in irreversible solidification [9]. Once the liquid photopolymer solidifies, it cannot return to its liquid state.

VP enables precise manufacturing at the micron level, making it suitable for producing precision parts or micro-devices. For instance, Saccone *et al.* [10] utilized VP to create metals and alloys with microscale resolution, a feat challenging to achieve using traditional processes. They infused 3D-architected hydrogels with metal precursors, which were then calcined and reduced to convert the hydrogel scaffolds into miniaturized metal replicas. This hydrogel-derived metal exhibits a highly twinned microstructure and remarkably high hardness, offering a pathway to

develop advanced metallic micromaterials (see Figure 2-2 i). In a study by Rafael *et al.* [11], micro-lattice structures produced by material extrusion and VP were compared, demonstrating the superior accuracy, quality, and relative density of the lattices manufactured by VP over material extrusion lattices (see Figure 2-2 ii).

Additionally, VP can solidify photosensitive materials directly into the desired shape without the need for molds, thus saving on manufacturing costs and time. VP encompasses various techniques such as stereolithography (SLA), digital light processing (DLP), and continuous digital light processing (CDLP). While all these are 3D printing technologies based on photopolymerization, they differ in their working principles, advantages, and disadvantages. In comparison with CDLP and SLA processes, DLP offers faster printing speed, higher resolution, and a simpler workflow. It enables the production of intricate structures and personalized customizations.

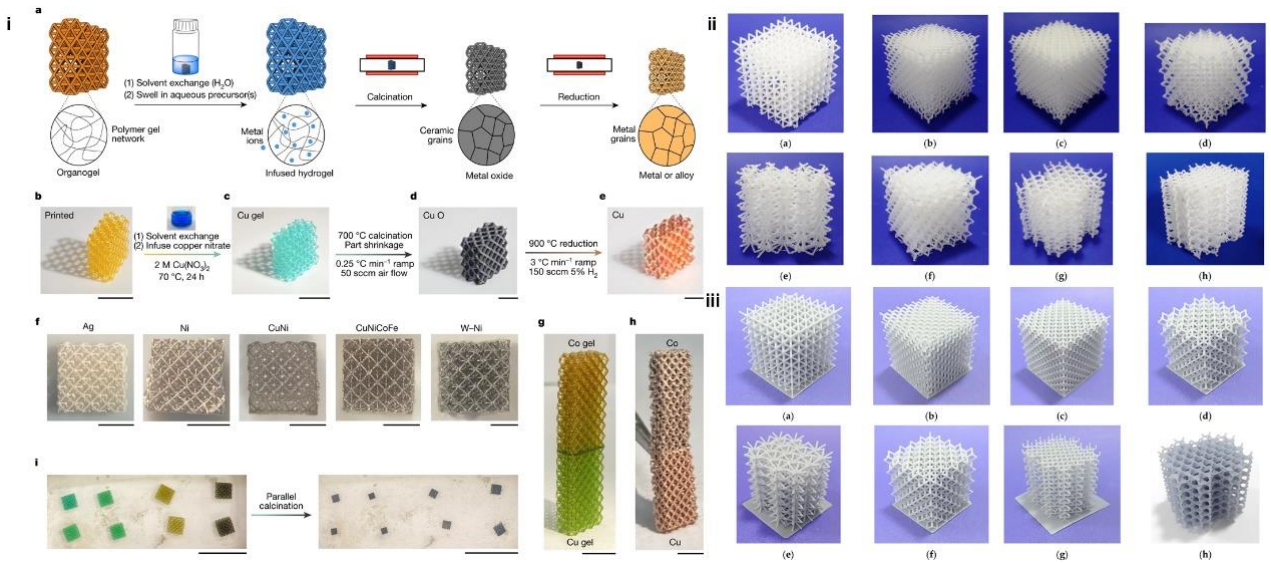


Figure 2-2 Research on the production of micro-devices by VP. (i) Schematic of hydrogel infusion additive manufacturing process to fabricate metal micro lattices [10] (ii) The lattice structures fabricated using ME and (iii) the Lattice structures built by VP [11].

2.1.1.1 Digital Light Processing

DLP technology is a rapid prototyping method that utilizes a projector to solidify photosensitive polymer liquid layer by layer, resulting in the creation of 3D printed objects. This molding process involves using slicing software to divide the model into thin slices, which are then projected by the projector. Each layer of the image triggers a photopolymerization reaction,

causing solidification in the area where the resin layer is extremely thin, forming a thin layer of the part. Subsequently, the molding platform moves one layer, and the projector continues to display the next slide, repeating the process for each subsequent layer until the printing is complete. This approach ensures high molding accuracy and rapid printing speed.

DLP offers a balanced approach to printing accuracy and cost-effectiveness. It excels at manufacturing objects with intricate structures, such as microfluidic structures [12], hollow sports shoes [13], and braces molds [14], at a relatively low cost (see Figure 2-3).

Furthermore, DLP possess unique advantages over other 3D printing technologies. Its stationary beam minimizes vibration deviation and eliminates issues like movable nozzles and material blockages, enhancing electrical safety. Additionally, DLP requires minimal printing preparation time, saving energy, and consumes less initial material, reducing user costs.

Moreover, DLP is capable of producing fine parts with sub-micron resolution, making it suitable for use in medical [15], jewelry [16], education [17], and other fields. While DLP technology is not intended for large-scale printing, it is commonly utilized in desktop-level 3D printers. Its versatility is evident in commercial applications, ranging from shoe soles [18] to COVID-19 testing swabs [19, 20] (see Figure 2-3).

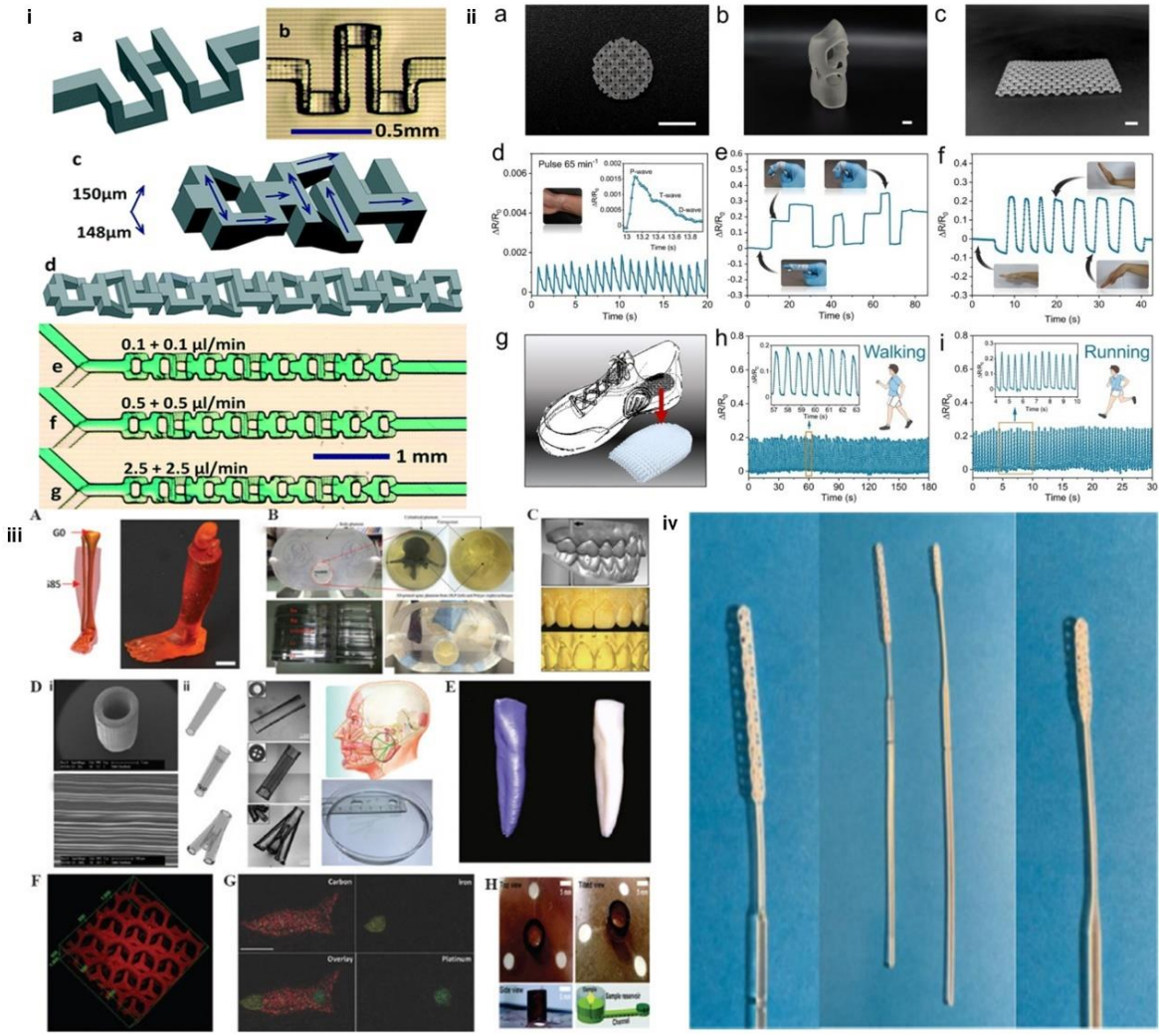


Figure 2-3 DLP printing applications. (i) particle filters printed by DLP [12]. (ii) DLP printed various customized structured flexible sensors for real-time detection of human movement [13]; (iii) DLP 3D printed medical devices [15] and (iv) DLP printed nasopharyngeal swabs for diagnosis of COVID-19 [20].

Photopolymer

The DLP-based 3D printing process involves the AM of molecules containing acrylate or acrylamide groups, which are activated by light through free radical initiation [21]. This method primarily utilizes photosensitive resin as the printing material, a material commonly used in SLA printing. The key components of photosensitive resin include small molecule resin (prepolymer, monomer), along with photo initiators, polymerization inhibitors, leveling agents, and other additives. These components undergo a polymerization reaction under specific light, typically ultraviolet light of a specific wavelength, to achieve curing [22].

The photosensitive resin used in DLP technology is consistent with SLA technology, with the

main ultraviolet light bands for curing being 405nm and 365nm. The materials are often modified with photosensitive resin as the substrate, offering various properties such as casting performance, short-term high temperature resistance, and good mechanical properties. Depending on specific requirements, the appropriate color ratio and transparency can also be selected as needed.

Photosensitive resin is not a new material. Comparable principles are extensively employed in electronic manufacturing, holographic imaging, adhesives, printing, medical uses, and various other fields. In the realm of coatings, DLP is recognized as an environmentally friendly technology due to its fast curing speed, excellent curing performance, minimal pollution, and energy efficiency [23]. However, the cured thickness of resin used in 3D printing (typically >25 μm) is significantly greater than the coating thickness of traditional coatings (typically <20 μm) [24]. Additionally, its formula composition differs from that of traditional light-curing coatings and inks.

Based on the polymerization system, photosensitive resin can be categorized into free radical polymerization and cationic polymerization, each relying on different polymerization mechanisms and reactive groups. Free radical polymerization utilizes the unsaturated double bonds in the photosensitive resin for polymerization, while cationic polymerization relies on the epoxy groups in the resin. The free radical system offers fast curing speed and cost-effective raw materials, but can be affected by the oxygen inhibition effect in the air, impacting curing and part performance. On the other hand, the cationic system is not affected by oxygen inhibition and exhibits minimal or no curing shrinkage but is sensitive to moisture and has higher raw material costs. As a result, the photosensitive resin predominantly used in 3D printing is based on the free radical polymerization system [25].

DMD chip

DLP printing employs a UV projector as a light source and operates by managing the projected light through a digital micromirror device (DMD). The DMD chip, a crucial component of DLP printing, is an optical semiconductor module featuring numerous minuscule lenses on its surface, each capable of independent flipping. This chip enables digital processing and light projection, manipulating the deflection of light through micro lenses to either reflect or not

reflect light. By integrating light sources and optical devices, DMD achieves binary patterns that surpass other spatial light modulation methods in terms of speed, precision, and efficiency. Each lens of the DMD can be individually tilted by +/- 12° around the hinged oblique axis. The deflection of the lens is controlled by altering the binary state of the underlying CMOS control circuitry and the lens reset signal, allowing it to tilt toward (on) or away from (off) the light source in the DLP projection system, resulting in bright or dark pixels on the projected surface [26].

The DMD chip's "mirrors" represent pixels, but due to their square nature, printing round or angular features at the pixel limit can result in a staircase effect, making it challenging to print borders smaller than 1 pixel. This limitation impacts printing accuracy.

Fidelity optimization

DLP printing involves complex interactions of chemical and physical processes, including light propagation, chemical reactions, species diffusion, heat transfer, volume shrinkage, and changes in mechanical properties of the curing resin[27]. Non-uniform light distribution during the curing process can lead to varying light intensities in different regions, affecting resin curing speed and uniformity. Additionally, non-uniform concentration gradients arising from chemical reactions or species diffusion due to uneven curing or light absorption can result in inconsistent chemical concentrations in specific areas, impacting printing precision and quality[1, 28, 29].

The limitations of achieving high accuracy in DLP printing become particularly pronounced when fabricating structures that demand exceptional precision, such as microchannels. Researchers have proposed addressing this challenge by implementing optical proximity corrections (OPCs), which involve adding sub-resolution assist features (SRAFs) or modifying the projected pattern[30]. Liu et al.[31] suggested using a grayscale projected pattern instead of a binary projected pattern and adjusting light distribution based on DMD to improve the alignment between the desired shape and the actual output. Just as in the studies by Lee et al.[32], Park et al.[33], and Mostafa et al.[34], researchers enhanced the resolution of their components by implementing grayscale adjustments to the projected images. However, these methods often require manual grayscale adjustments, relying heavily on trial and error, which may not be suitable for complex geometries. A recent technique by Zhou et al.[35] utilizes linear

programming optimization to minimize boundary errors and attain sub-pixel resolution, focusing on refining individual layer boundaries and employing a simplified curing model. Nonetheless, the sheer number (255) and arrangement possibilities of grayscale pixel values pose a significant challenge (see Figure 1-1 i). Therefore, a key challenge lies in determining the optimal allocation of grayscale pixels.

In recent DLP printing advancements, we have seen the emergence of grayscale DLP (g-DLP) printing, which allows for the creation of structures with locally varying mechanical properties through the control of light intensity. For instance, Montgomery et al. [1] utilized a reaction-diffusion model to predict the shape of printed objects before and after grayscale augmentation. It developed a numerical representation to simulate the light pattern of pixels in the DMD array, which is crucial for studying the relationship between overcuring and pixel grayscale. This model, informed by experimentally measured light distributions that mirror the pattern generated by the DMD device in the UV projector, is capable of predicting the form of smoothly rounded features as small as 3.5 pixels, equivalent to a diameter of 352 µm (see Figure 2-4 i). KUANG et al. [36] developed a novel two-stage curing hybrid ink system for single-vat g-DLP 3D printing, with applications in 2D and 3D lattices, cellular structures, and metamaterials (see Figure 2-4 ii). Mostafa et al. [34] manipulated voxel dimensions by controlling three parameters of grayscale pixels, resulting in improved geometric dimensions and tolerances of printed parts. Additionally, Forte et al. [37] introduced a method for designing modulus distributions by leveraging finite element analysis (FEA) with g-DLP printing. This innovative approach enables the fabrication of unique modulus distributions, facilitating the accurate production of complex parts, including those with traditionally perceived defect features like holes and sharp corners. The average stress-strain curves of the improved and neutral models, in contrast to their equivalent uniform models, demonstrate an 85% increase in failure strain for the improved model (see Figure 2-4 iii). However, these models typically necessitate measurements of ink solidification or mechanical properties, necessitating the inclusion of additional time-consuming characterization steps in the workflow. Furthermore, processing extensive data sets often mandates the use of high-performance computers or parallel computing clusters, thereby amplifying the complexity of the overall process. These factors, combined with the need for sophisticated experimental setup and specialized equipment, contribute to the significant

experimental cost associated with these modelling methods.

Utilizing the grayscale function of each pixel to compensate for distortion is a promising approach to enhancing print quality. Zhao *et al.* [38] developed an optimization method for g-DLP printing of rectangular blocks using machine learning (ML) and evolutionary algorithms (EA) (see Figure 2-4 iv). They trained a recurrent neural network (RNN) model using a database formed by the FEM model to predict deformation shapes quickly and accurately, leading to optimal design and demonstrating the effectiveness of this ML approach.

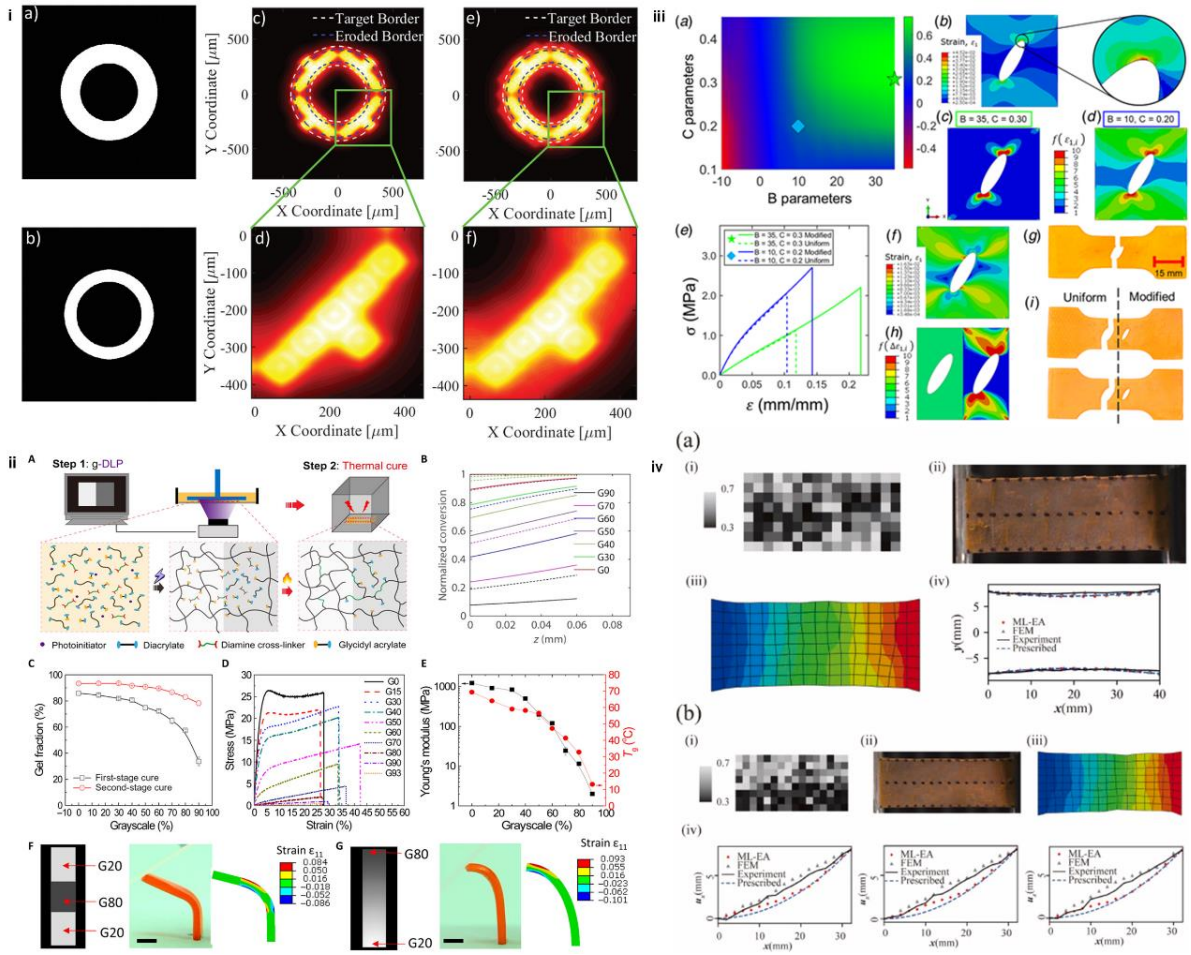


Figure 2-4 Recent research progress on grayscale DLP printing. (i) The grayscale smoothing procedure to design a grayscale pattern that cures a shape most closely matching a target image [1]. (ii) g-DLP 3D printing of FGM via two-stage curing [36]; (iii) Simulating and printing rotating elliptical holes to demonstrate the mechanical properties of g-DLP printed materials [37] and (iv) Optimal design of grayscale gradient using ML-EA method [38].

g-DLP printing enhances printing precision and surface quality, while also broadening the spectrum of printable materials. Through precise control of gray levels, finer printing resolutions can be attained, resulting in intricately detailed and smooth surfaces on printed

objects. Furthermore, g-DLP printing enables multi-material printing capabilities. By adjusting the blend ratio of light-curing resin, structures with varying hardness, transparency, and other unique properties can be produced to cater to diverse application requirements.

2.1.1.2 Other Vat Photopolymerization Methods

SLA technology, the earliest 3D printing method, is widely used in industry due to its ability to produce precise and flexible designs using liquid photopolymer resin. For example, Wang *et al.* [39] demonstrated its potential in manufacturing drug-loaded tablets with specific extended-release profiles, suggesting its suitability for oral dosage forms, industrial production, and personalized dosage (see Figure 2-5 i). Similarly, Chen *et al.* [40] utilized SLA printing to create complex-shaped scaffolds with hydroxyapatite powder, showcasing its potential for biosafety implants (see Figure 2-5 ii). Despite its maturity, SLA technology has drawbacks, including material flexibility, high cost, and low strength of printed objects.

Continuous Digital Light Processing (CDLP) is a novel 3D printing method utilizing liquid photo-curable resin to construct layered structures. Its benefits include rapid printing, high resolution, and the ability to create intricate designs. For instance, Wallace *et al.* [41] demonstrated CDLP's capability to produce precise patient-specific scaffolds and intricate pore geometries suitable for cell interaction (see Figure 2-5 iii). However, CDLP also has drawbacks, such as increased material and equipment costs, as well as potential emission of toxic gases during photocuring. As technology advances, these limitations may be addressed or minimized.

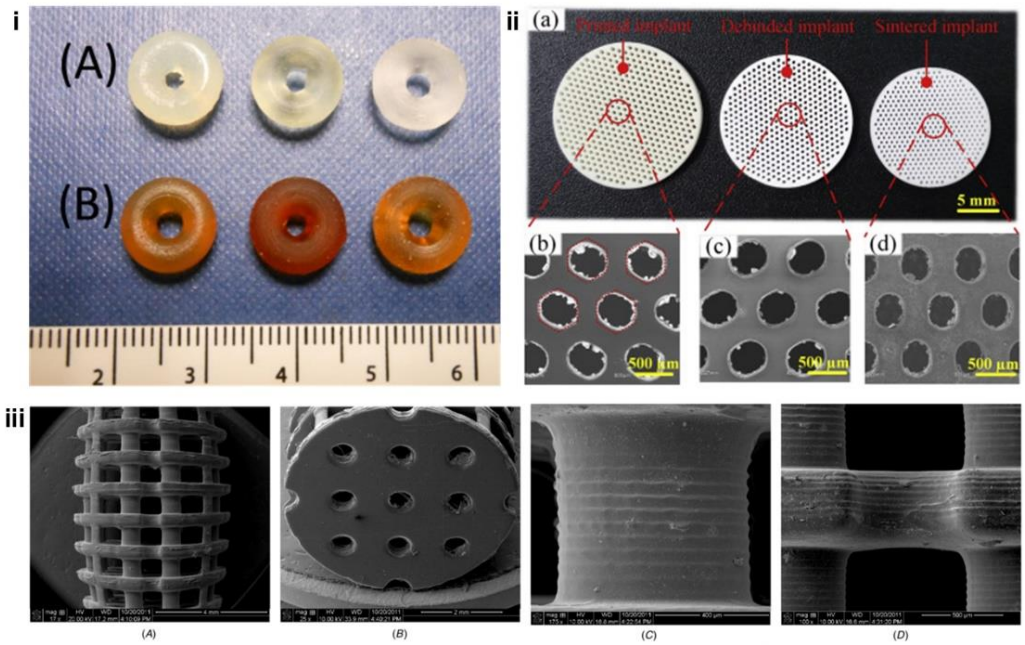


Figure 2-5 Applications of other VP methods. (i) Drug loaded tablets were fabricated with the SLA printer [39]. (ii) The HAP implants with micro-holes was manufactured by SLA-3D printing [40]; (iii) CDLP additive manufacturing of full-size scaffolds [41].

2.1.2 Basic Additive Manufacturing Methods Beyond Vat Photopolymerization

In addition to VP, other AM methods have their own advantages and limitations, often related to cost, speed, material properties, and geometric constraints. For instance, material extrusion is widely used for rapid and precise manufacturing by extruding materials into specific shapes, making it versatile and suitable for producing parts, prototypes, handicrafts, and biomaterials (see Figure 2-6 a). On the other hand, material jetting builds parts layer by layer by ejecting droplets of photosensitive material and hardening them with ultraviolet light, enabling the production of smooth-surfaced, dimensionally accurate parts (see Figure 2-6 b). Similarly, binder jetting sprays binder to hold powdered materials in place and bond building components layer by layer, making it suitable for 3D printing of powder materials such as metals and ceramics. This method is capable of producing parts with good mechanical properties for complex metal parts and ceramic products [42] (see Figure 2-6 c).

Apart from these methods, AM encompasses three other fundamental techniques. Powder bed fusion evenly spreads a thin layer of powder material on a work platform, then utilizes a laser or electron beam to melt or sinter specific areas, forming a solid part. This process is suitable for 3D printing of metals, plastics, and ceramics, widely used in aerospace, automobile

manufacturing, medical equipment, and other fields (see Figure 2-6 d). Directed energy deposition technology, commonly used in high-tech metal industries, involves depositing metal powder onto a build platform and then melting it to form a solid structure (see Figure 2-6 e). Sheet lamination entails stacking layers of paper or plastic sheets, each coated with a binder, and cutting them to achieve the desired part profile (see Figure 2-6 f). While this method reduces manufacturing costs and enables the production of larger-sized parts, it may result in a less smooth surface quality and present challenges for manufacturing complex geometries [43].

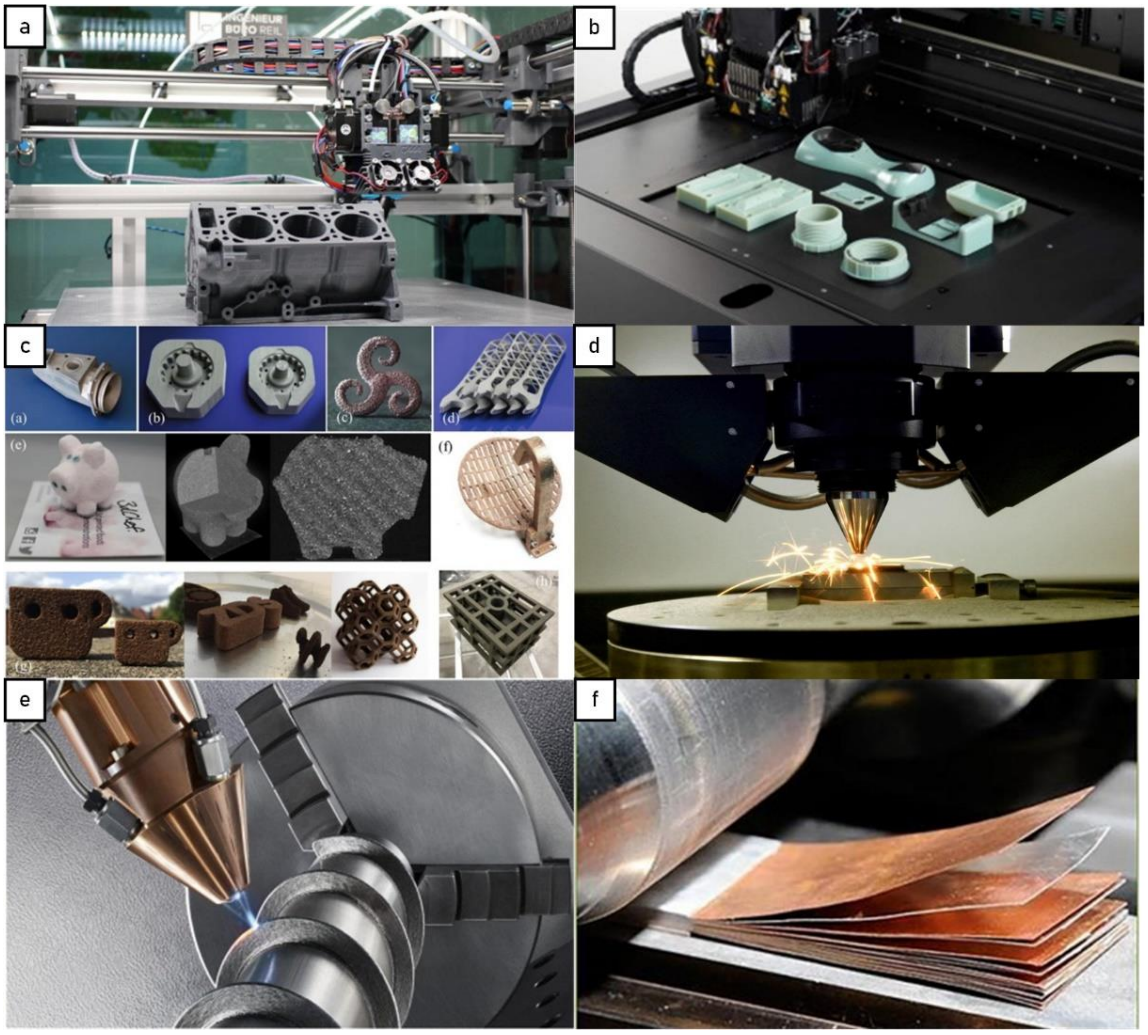


Figure 2-6 Different Additive Manufacturing Processes. (a) Material extrusion process [43]; (b) Developed product through material jetting [43]; (c) Parts produced by binder jetting [42]; (d) Powder bed fusion process; (e) Direct energy deposition process and (f) Sheet laminated process[43].

2.2 Machine learning

ML, a branch of artificial intelligence (AI), allows computers to learn from data to recognize patterns, make predictions, and decisions. It involves training algorithms to improve their performance without explicit programming. By analyzing large amounts of data, ML algorithms can identify patterns and create predictive models for new data. ML has diverse applications in image recognition, natural language processing, recommendation systems, and financial forecasting, making it crucial in modern technology and AI. ML is roughly divided into four main types: supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning, each with its own advantages and disadvantages [44].

Supervised learning is the most commonly used ML method, where a machine is trained on labeled data to predict outputs based on the provided data. In contrast, unsupervised learning involves training the machine on unlabeled data to predict outputs without supervision, grouping data based on similarities and patterns. Semi-supervised learning combines characteristics of both supervised and unsupervised ML by using a mix of labeled and unlabeled datasets to train its algorithm. Reinforcement learning is a feedback-based process in which the AI component assesses its surroundings, takes actions, learns from experience, and aims to maximize rewards by executing good actions, being rewarded for positive outcomes and punished for negative ones.

Vertical industries dealing with substantial data volumes increasingly appreciate the significance of ML technology. Real-time insights derived from ML facilitate efficient operations and confer a competitive edge. Presently, ML is predominantly employed in healthcare, banking, travel, and social media. Nonetheless, traditional ML methods necessitate manual feature engineering, exhibit constrained processing capabilities for intricate data and nonlinear relationships, and struggle with generalizing to new data.

2.2.1 Deep learning

Deep learning (DL) is a supervised learning approach within ML that aims to achieve better performance than traditional ML algorithms by replacing model specifications with hierarchical features. DL learns the inherent laws and representation levels of sample data by training on

input-output pairs, focusing on feature representation rather than feature engineering to create adaptable, generalizable systems that can continuously improve with new data, marking a paradigm shift in machine learning [45].

Computer vision (CV) is a crucial aspect of DL that allows computers and systems to extract valuable information from images, videos, and other visual inputs, enabling them to take actions or provide recommendations based on this information. The three primary tasks in CV are image classification, object detection, and image segmentation [46].

2.2.1.1 Image classification

Image classification, a subcategory within CV, involves categorizing input images into different classes. Common methods for image classification include traditional ML algorithms like support vector machines (SVM), K-nearest neighbors (KNN), and random forests (RF), as well as DL algorithms such as convolutional neural networks (CNN). DL excels in image classification tasks as it can automatically learn image features without manual feature extraction.

Image classification is widely used in various fields, including medical imaging diagnosis, autonomous cars, security monitoring, image search, and social media. For instance, Possatti *et al.* [47] utilized DL to accurately detect traffic lights along the trajectory, addressing the challenge of traffic light detection for autonomous vehicles (see Figure 2-7 a). This technology enables computers to automatically identify image content, enriching user experiences. As DL continues to advance, the accuracy and efficiency of image classification are consistently improving, offering new possibilities across various domains.

2.2.1.2 Object detection

Object detection, aims to identify and locate specific objects or targets in images or videos. Unlike image classification, object detection not only involves identifying object categories but also determining their precise location through bounding boxes. Leveraging ML and DL techniques such as CNN and RNN, these algorithms excel at handling objects of varying scales, shapes, and quantities, making object detection widely applicable in intelligent transportation, security monitoring, autonomous driving, and medical image analysis.

With the continuous development of DL, the precision and effectiveness of object detection are consistently enhancing, providing crucial technical backing for intelligent and automated systems. For instance, in their work on CenterNet, Zhou *et al.* [48] introduced an end-to-end detection network based on keypoint detection, capable of integrating 3D object detection, human pose estimation, optical flow learning, depth estimation, and other tasks within a unified framework (see Figure 2-7 b).

2.2.1.3 image segmentation

Image segmentation, a critical task in CV, aims to divide images into semantically meaningful regions or objects. Unlike object detection, it not only involves identifying objects in an image but also accurately labeling each pixel with the object or region it belongs to. This technique is utilized in various fields, including medical image analysis (such as organ segmentation and lesion detection), autonomous driving (road and obstacle recognition), land parcel classification, and industrial quality control and product inspection. For instance, Wang *et al.* [49] proposed an efficient interactive 2D/3D medical image segmentation framework based on DL. Experiments on segmenting multiple organs from 2D fetal MRI and segmenting brain tumors from 3D MRI demonstrate that this method requires less user interaction, while achieving similar or better accuracy in less time (see Figure 2-7 c). Recently, DL-based image segmentation methods have achieved remarkable results, enhancing the accuracy and efficiency of segmentation.

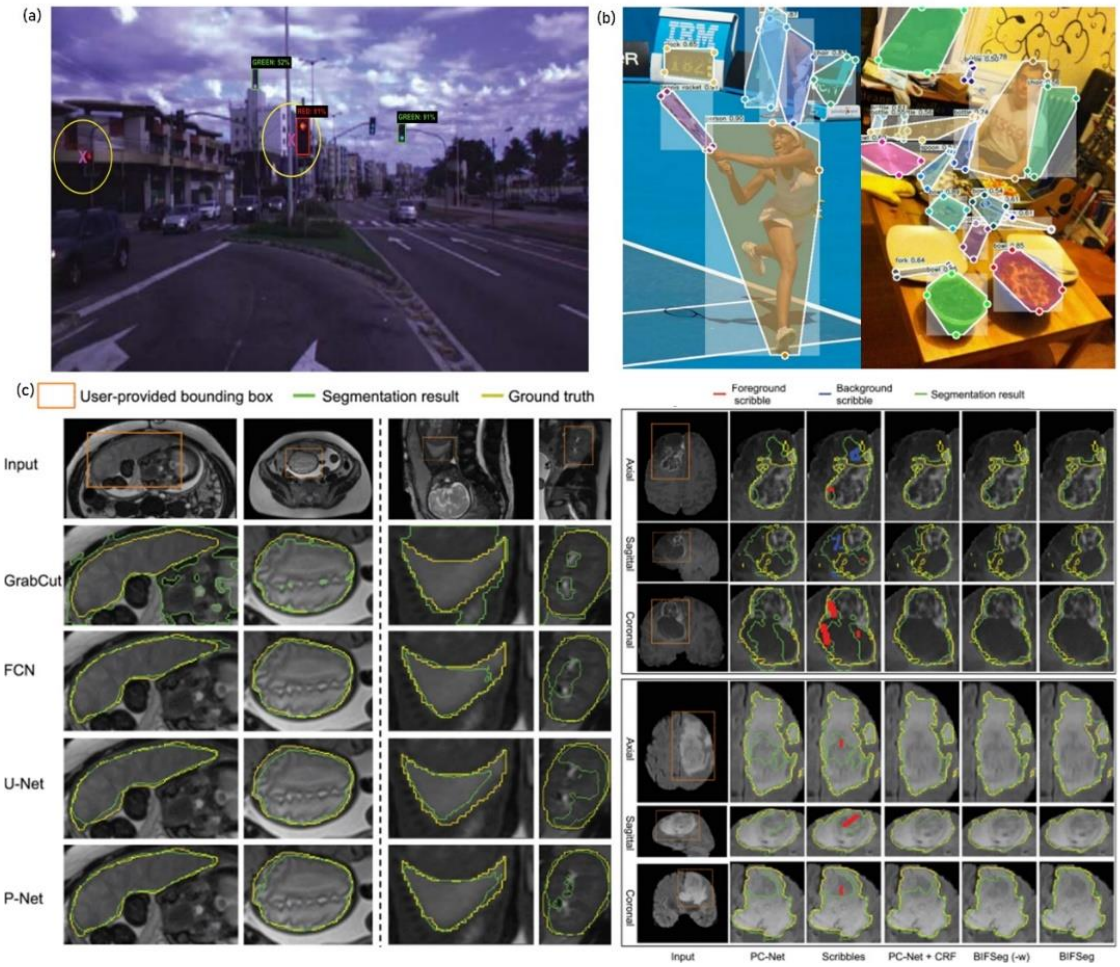


Figure 2-7 Three classic tasks of CV. (a) The identification and recognition of traffic lights using a deep detector in an online phase, with the lights delimited by bounding boxes [47]. (b) Objects are detected by locating their extreme points, which are then used to form a bounding box, providing a tighter octagonal approximation of the object [48]. (c) Compares various DL models for segmenting multiple organs in 2D fetal MRI and brain tumors from a 3D bounding box [49].

U-Net

U-Net is a classic DL architecture for image segmentation, proposed by Ronneberger *et al*. [50] in 2015. It is named after its U-shaped network structure (see Figure 2-8). U-Net is commonly employed for semantic segmentation of medical images, satellite images, *etc*., such as segmenting organs and diseased areas. This model excels in image segmentation tasks by effectively segmenting images while preserving high resolution.

The structural features of U-Net include a symmetrical encoder-decoder structure and skip connections, which contribute to its strong performance in image segmentation tasks. In the encoder part, U-Net progressively reduces the size and channel count of the feature map using convolution and pooling layers to extract high-level features from the image. Conversely, the

decoder part gradually restores the size and channel count of the original image through upsampling and convolution layers. Simultaneously, skip connections are employed to merge the features extracted in the encoder with those in the decoder, aiding in the preservation of spatial information and enhancing segmentation accuracy.

U-Net also excels in grayscale prediction tasks. To utilize U-Net for grayscale prediction, the initial step involves preparing training data, which includes input images and their corresponding label images. Subsequently, DL frameworks such as TensorFlow, PyTorch, *etc.*, can be employed to implement the U-Net model, define the network structure, loss function, and optimizer. Following this, training on the prepared data allows for the optimization of the parameters of the U-Net model, enabling it to learn how to predict grayscale images from input images. Upon completion of the training process, the model becomes capable of predicting new grayscale images. For instance, Francesco *et al.* [51] successfully integrated U-Net and its variant algorithms to predict the average muscle ultrasound grayscale value in muscle ultrasound images acquired at different times.

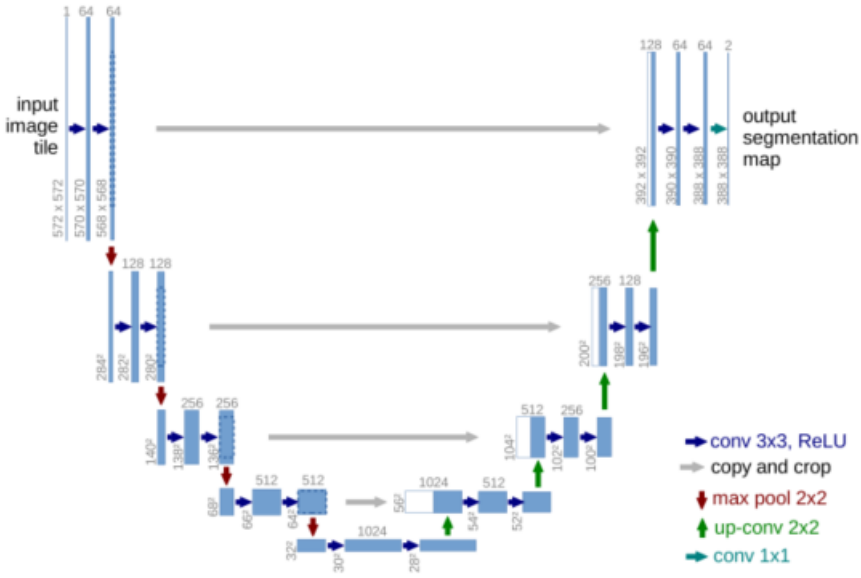


Figure 2-8 U-Net architecture [50]

Related segmentation algorithms

Apart from U-Net, there exist numerous other image segmentation algorithms, such as Mask R-CNN, Fully Convolutional Network (FCN), and DeepLab.

Mask R-CNN is a DL-based image segmentation algorithm that combines object detection and semantic segmentation. It is well-suited for tasks that require simultaneous recognition and localization of objects. For instance, Anantharaman *et al.* [52] employed Mask R-CNN to detect and segment oral diseases, while Chiao *et al.* [53] utilized Mask R-CNN to detect and classify breast tumors in ultrasound images.

Fully Convolutional Network (FCN) is an end-to-end image segmentation algorithm suitable for various scenarios. For example, Li *et al*. [54] utilized a deep FCN for pixel-level land and sea segmentation, and Bian *et al.* [55] applied a multi-scale FCN for segmenting and quantifying the spalling of aircraft engine blades. DeepLab is a semantic image segmentation algorithm that utilizes deep CNN and atrous convolutions. It is used for parcel segmentation, medical image analysis, and environmental perception. For example, Wang *et al.* [56] applied DeepLab for high-precision recognition and segmentation of gastric cancer pathological slice medical images, while Venugopal [57] used DeepLab dilated learning networks for automatic semantic segmentation to detect changes in remote sensing images.

These algorithms are suitable for medical image analysis, natural image segmentation, satellite image analysis, and other fields to extract areas of interest, identify objects, and understand scenes.

2.2.2 Adversarial learning

Generative adversarial networks (GANs) [58] represent a significant advancement in ML. GANs can create valuable data from scratch or random noise, typically in the form of images. Essentially, GANs consist of two neural networks: a generator and a discriminator. The generator aims to produce data that closely resembles real samples, while the discriminator seeks to differentiate between real data and that generated by the generator. These networks compete during training, enabling the generator to produce data so realistic that the discriminator cannot distinguish between real and generated data. In essence, GANs involve two neural networks competing to find the best path. GANs have been successfully utilized in image generation, style transfer, image enhancement, and various other fields.

2.2.2.1 Conditional Generative adversarial networks (CGANs)

CGANs expands upon the capabilities of standard GANs. In CGANs, the generator and discriminator are trained not only to create and differentiate real data, but also to do so based on external conditions. This additional information can take various forms, such as labels or text descriptions. By incorporating these conditional factors, CGANs can produce data in a more targeted manner, such as generating images based on specific labels[59]. CGANs have demonstrated strong performance in tasks such as image generation and editing.

Antipov *et al.* [60] employed CGANs for facial aging and objectively evaluated the resulting aged and rejuvenated face images with state-of-the-art face recognition and age estimation solutions, demonstrating the significant potential of CGANs (see Figure 2-9 a). Another notable application is the use of CGANs for shadow detection by Nguyen *et al.* [61]. Previous shadow detection methods have primarily focused on learning the local appearance of shadow regions, utilizing limited local context in the form of pairwise potentials in conditional random fields. In contrast, CGANs can model higher-level relationships and global scene features, capable of generating shadow maps corresponding to different sensitivity levels, eliminating the need for multiple models and expensive training procedure (see Figure 2-9 b).

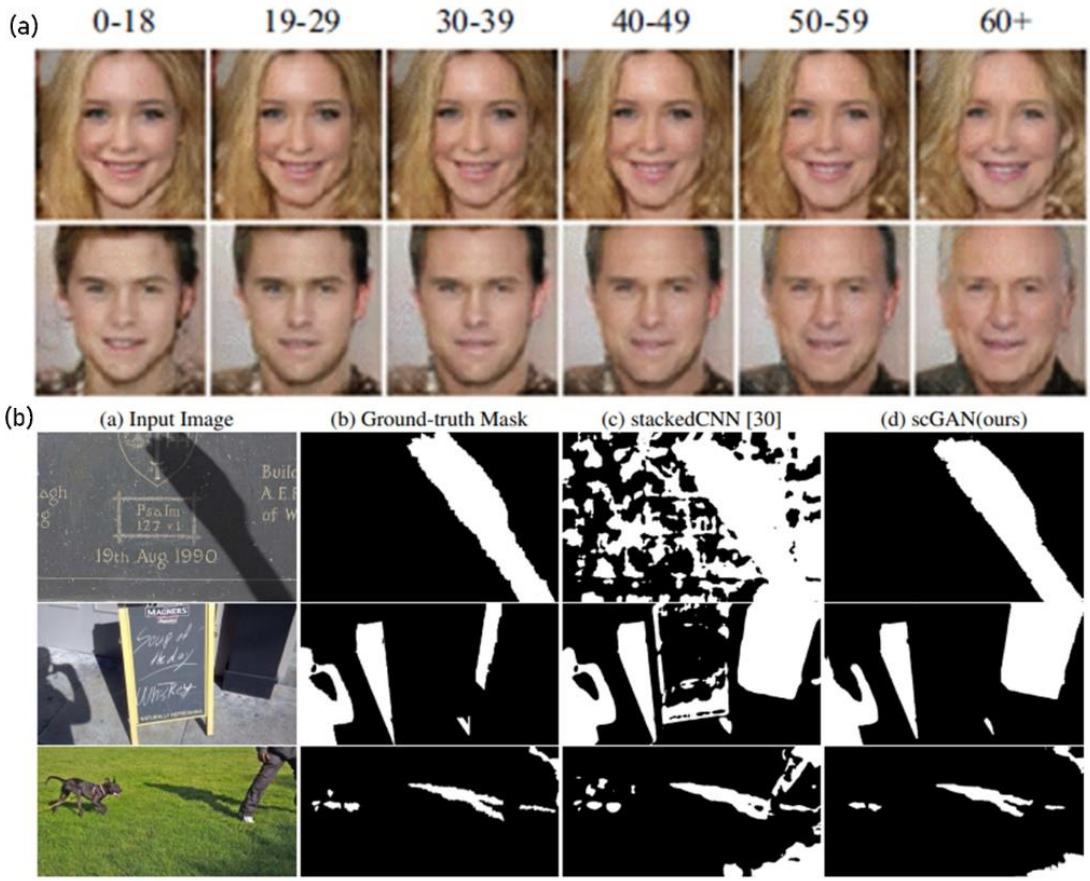


Figure 2-9 The powerful performance demonstrated by CGANs. (a) Synthetic faces of different ages generated with CGANs [60]; (b) Comparison of the performance of CGANs and traditional CNN in qualitatively detecting shadows [61].

Pix2Pix

Pix2Pix is an image translation model based on CGANs, a variant of GANs that can be constrained by input images or labels when generating images. The Pix2Pix model utilizes the CGANs architecture to produce the desired output image using the input image as a condition, such as converting a black and white photo into a color image or a sketch into a realistic photo. It achieves this translation by learning the relationship between the input image and the corresponding output image. Pix2Pix has been successfully applied to various image translation tasks. For example, Qu *et al.* [62] employed the Pix2Pix network for dehazing, generating haze-free images without relying on physical scattering models. Similarly, Salehi *et al.* [63] utilized a Pix2Pix-based stain-to-stain translation method for stain normalization of Hematoxylin and Eosin (H&E) - stained histopathology images. The flexibility and efficiency of Pix2Pix have made it a highly regarded research direction in the field of image processing, with broad application

prospects in CV and image processing, providing powerful tools and technical support for image processing tasks.

2.2.2.2 Unconditional Generative Model

Unconditional generative models are instrumental in generating new data samples from a given dataset without specific conditions or constraints. The foundational GANs is a classic unconditional generative model, widely applied in tasks such as image generation, text generation, and music composition, contributing to the creation of novel and distinctive content.

In addition to GANs, another popular unconditional generative model is the variational autoencoder (VAE). VAE is a type of neural network trained to encode and decode data samples, learning a probability distribution of the data and generating new samples by sampling from this distribution. This makes VAE suitable for tasks such as image synthesis, data augmentation, and anomaly detection. For instance, Razavi *et al.* [64] utilized VAE to generate diverse high-fidelity images, while Rampasek *et al.* [65] proposed a drug response variational autoencoder to improve the accuracy of drug response prediction.

Unconditional generative models are also utilized in creative fields such as art and music, continually being refined and improved to generate new and unique content.

2.3 Machine Learning and Additive Manufacturing

The integration of ML and AM is a compelling area of exploration. As AM technology continues to progress, the utilization of ML algorithms has become a focal point within this domain. ML technology has the potential to enhance AM processes, elevating efficiency and product quality by analyzing extensive data patterns. For instance, ML can predict material properties, optimize process parameters, enable more complex part designs, and refine manufacturing processes through real-time monitoring and feedback. Additionally, ML can aid in identifying and addressing defects and issues in manufacturing, offering intelligent solutions. Consequently, the fusion of ML and AM presents both opportunities and challenges to the manufacturing industry, propelling the further evolution of AM technology.

2.3.1 Machine Learning Development in DLP 3D Printing

DLP 3D printing is known for its high precision, efficiency, and design freedom as a rapid prototyping technology. However, challenges arise in its manufacturing process, such as optimizing the formulation of light-curing resin and addressing potential defects and issues during printing. Integrating ML technology with DLP 3D printing can drive innovation and development within the manufacturing industry, leading to improved processes and outcomes and unlocking new opportunities for advancement.

2.3.1.1 Academic research on the integration of ML and DLP 3D printing

Research on the integration of ML and DLP 3D printing primarily encompasses optimizing light-curing resin formulations, refining printing parameters, enhancing printing accuracy, predicting potential defects and issues during the printing process, as well as implementing automated control and intelligent monitoring throughout the printing process.

Optimization of Light-Curing Resin Formulations

The inadequate performance of current materials has impeded progress in utilizing 3D printing for complex structure manufacturing and customized designs. Traditional trial-and-error methods focus on a single objective, resulting in high experimental costs. The application of ML techniques to accelerate the discovery of high-performance 3D printing materials with minimal experimental expenses has become a significant research area. ML research in DLP resin formula optimization primarily involves enhancing formulations, selecting materials, and optimizing production parameters. Utilize ML algorithms to analyze the impacts of different components in DLP resin, optimizing the formulation, selecting suitable materials, and adjusting production parameters to enhance product quality and efficiency while effectively reducing costs. For example, Gao *et al.* [66] developed an efficient ML workflow using an active optimization approach to identify a range of 3D printing materials with exceptional mechanical properties through minimal experiments. Albuquerque *et al.* [67] employed ML to design bio-based multi-component epoxy resin system formulations, successfully creating epoxy resin systems with tailored properties.

Applying ML techniques to optimize resin formulas speeds up the optimization process, saves

time and labor costs, and fosters the development and innovation of new products, ultimately contributing to the attainment of sustainable development objectives.

Refinement of Printing Parameters

In the field of 3D printing, identifying the optimal printing parameters has consistently posed a significant challenge. However, the utilization of ML technology has introduced innovative solutions to address this issue. Through the collection of comprehensive data during the printing process and the application of ML algorithms to analyze and identify the key parameters that impact print quality and efficiency, we can precisely fine-tune these parameters to achieve superior printing results.

Dabbagh *et al.* [68] demonstrated the integration of ML and 3D printing by developing a user-friendly GUI that allows users to assess the impact of printing parameters on print quality. This approach reduces pre-printing uncertainties by identifying optimal printing parameters to regulate process time and material usage, eliminating the need for trial and error. In a similar vein, Shirmohammadi *et al.* [69] utilized a hybrid artificial neural network model and particle swarm algorithm to optimize 3D printing process parameters, aiming to minimize surface roughness and enhance the surface quality of printed components throughout the printing process.

ML facilitates the automatic adjustment of printing parameters based on material properties and printing requirements, enabling personalized customization and enhancing the adaptability and versatility of printing processes. This advancement opens up new possibilities for the progression and broadened application of 3D printing technology.

Enhancing Printing Precision

As previously discussed, g-DLP printing represents a significant advancement in enhancing the accuracy of DLP printing. Traditional commercial 3D printers utilize binary black (minimum light output) and white (maximum light output) digital masks, limiting the printer's capacity to adjust exposure for intricate voxel interactions. Leveraging the full bit depth of the light engine's grayscale mask can extend the capabilities of VP. The utilization of grayscale levels allows for a more nuanced control over the exposure, with gray pixels exhibiting lower light intensity than

white pixels, leading to a reduced reaction rate relative to voxel size at the same exposure duration. Consequently, grayscale manipulation has emerged as a potent tool for optimizing mask design, particularly in smoothing out the edges of local features to eliminate the staircase effect and enhance feature precision.

VP entails a vast number of reactions and interactions within a single component, making it well-suited for applications involving big data, ML, and AI. Despite its inherent compatibility with ML modeling, only a limited number of studies have integrated ML tools into VP, with most approaches to g-DLP printing relying on physical modeling. By leveraging extensive, high-precision datasets, ML models can establish a correlation between input mask illumination and output voxel geometry. For instance, Killgore *et al.* [70]'s confocal microscopy-based workflow enables the rapid acquisition of voxel interaction data from random grayscale digital photomasks, employing techniques such as U-Net and GANs to enhance the utilization of grayscale digital light processing (see Figure 2-10 i). This enables more accurate voxel predictions in material manufacturing, thereby enhancing the precision of DLP printing.

Researchers like You *et al.* [2] have utilized deep neural networks (NN) based on ML technology to explore the intricate relationship between input digital masks and the resulting 3D printed structures, effectively mitigating scattering effects during the printing process (see Figure 2-10 ii). Similarly, Guan *et al.* [71] have employed DL methods to automatically generate printer parameters for compensating scattering effects, addressing light scattering challenges in multi-layer 3D bioprinting (see Figure 2-10 iii).

In general, g-DLP printing enhances the versatility and variety of printing capabilities, opening up new avenues for innovation in manufacturing and design industries. This advancement not only propels the progress of 3D printing technology but also broadens the horizons of its applications, ushering in a new era of possibilities for creators and manufacturers alike.

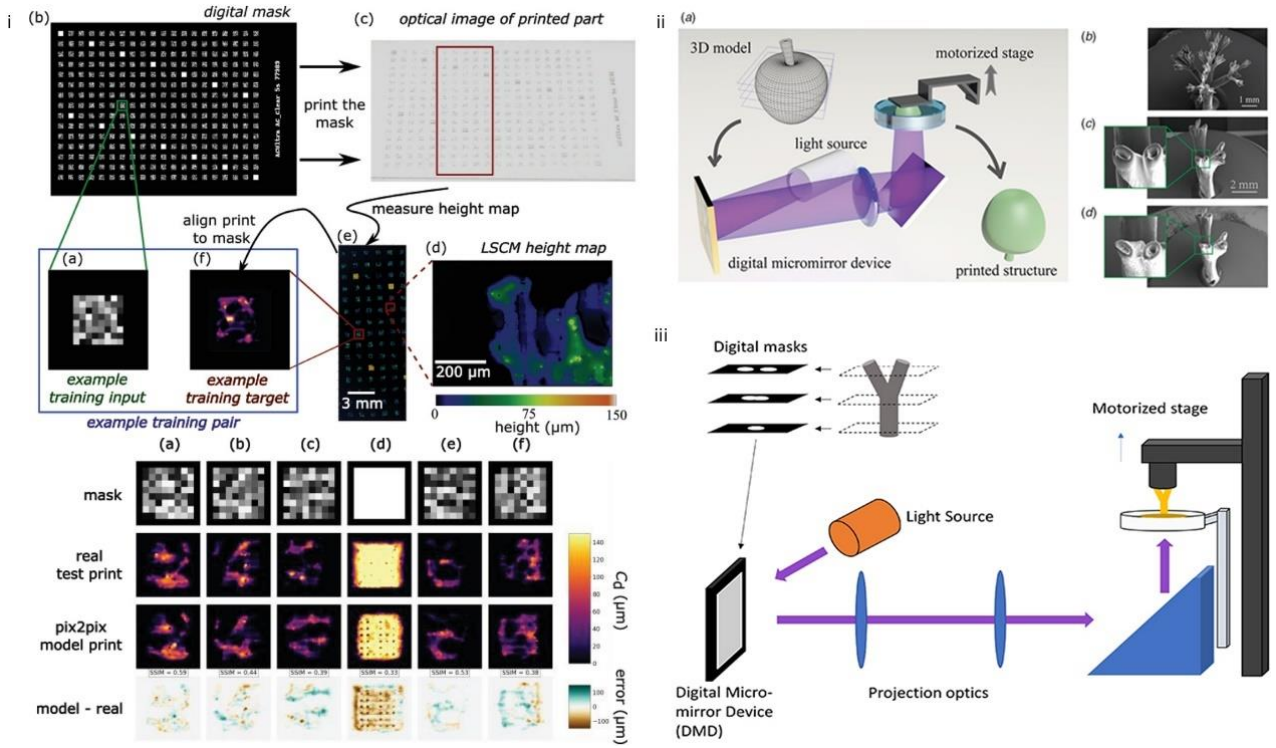


Figure 2-10 The latest development in g-DLP printing: (i) Procedure for capturing and analyzing grayscale mask and its corresponding printed voxel geometry data; Comparison between the actual test print and the predicted print [70]. (ii) SEM images of a printed 3D "fractal tree" and two versions of a "blood vessel" - one with a non-scattering material and the other with a scattering material, each showing a detailed zoom-in of the vessel opening [2] and (iii) Schematic of the DLP-based 3D bioprinting setup [71].

Automated Control and Intelligent Monitoring in DLP printing

Detecting defects in DLP printing minimizes material and time wastage. Early defect detection can trigger alerts to halt printing for corrective action, avoiding reprinting. Leveraging machine learning for quality monitoring enables real-time assessment, predictive maintenance, automatic adjustments, and data-driven decision-making, enhancing printing quality and production efficiency. For instance, Delli *et al.* [72] proposed a method integrating cameras, image processing, and supervised ML to automatically evaluate 3D print quality. By capturing images of semi-finished parts at key printing stages based on part geometry, defects can be promptly and effectively identified. Khan *et al.* [73] developed a real-time DL model to detect critical defects by extracting geometric anomalies in infill patterns, such as inconsistent extrusion or weak infill, comparing them with ideal prints to prevent production losses and reduce manual quality inspection. Analyzing and learning from extensive printing data through ML provides production managers with data-driven decision support, aiding in problem

identification, failure prediction, parameter optimization, and achieving enhanced production efficiency.

2.3.1.2 Commercial Intelligent DLP 3D Printer

The commercial intelligent DLP 3D printer is an advanced manufacturing device that combines ML technology with DL algorithms and data analysis to optimize the printing process and enhance print quality. These printers can automatically adjust parameters based on specific printing requirements, thereby improving production efficiency. Through ML, they can learn and optimize print paths, material usage, and print speeds to boost production efficiency and final product quality. Moreover, these printers can also automatically detect and prevent faults, reducing production downtime and enhancing the stability and reliability of the production line.

One of the prominent companies in this field is Carbon, a US-based technology firm specializing in digital manufacturing technology. Carbon's DLP 3D printers leverage ML technology to deliver high-quality 3D printing solutions across industries like automotive, medical, and aerospace. By integrating ML with advanced 3D printing technology, Carbon's products have made significant strides in printing speed, accuracy, and reliability, empowering customers with increased productivity and innovation.

In addition to Carbon, companies like Formlabs and Stratasys are also introducing intelligent DLP 3D printers that utilize ML technology to provide smarter and more efficient manufacturing solutions. Stratasys, a leading global 3D printing solutions provider headquartered in Israel and the US, integrates ML technology into their smart DLP 3D printers to enable intelligent printing process management and optimization, offering customers efficient and reliable manufacturing solutions. By continually exploring ML applications in 3D printing, Stratasys is dedicated to advancing intelligent manufacturing technology and delivering greater value and competitive advantages to customers.

The integration of ML with DLP 3D printing is poised to drive innovation, development, and broader adoption of this technology in manufacturing. These technology companies are at the forefront of exploring ML applications in 3D printing, advancing intelligent manufacturing technology, and presenting new opportunities and challenges to the global manufacturing

industry.

2.3.2 Machine Learning Development in various 3D Printing methods

The integration of ML with various 3D printing methods is a significant trend. Currently, the application of ML technology in metal 3D printing has yielded impressive outcomes. For instance, optimizing process parameters in metal 3D printing using ML algorithms can enhance printing quality and efficiency. Employing ML to analyze data during the printing process can proactively identify and address potential issues. Yang *et al.* [74] utilized ML to monitor melt pool dynamics in real-time during ultrasound-assisted metal 3D printing, thereby reducing defects and enhancing printing quality. Wang *et al.* [75] leveraged ML to develop an in-situ control system for enhancing the liquid metal jet printing process and achieving more stable jetting behavior.

Furthermore, ML plays a pivotal role in various processes such as laser sintering, inkjet printing, and electron beam melting. In laser sintering, ML can optimize parameter settings to improve sintering quality and efficiency. By analyzing data from the laser sintering process, ML algorithms can fine-tune parameters like laser power and scanning speed to achieve precise sintering results. In inkjet printing, ML can enhance the control and operation of inkjet printers. Through analysis of different inkjet materials and parameters using ML algorithms, printing quality and speed can be improved. Similarly, electron beam melting, a high-precision additive manufacturing process, can leverage ML to optimize parameter settings for enhanced printing quality and accuracy. By utilizing ML algorithms to analyze data from the electron beam melting process, more accurate printing outcomes can be achieved.

In summary, DLP 3D printing still faces challenges in achieving high accuracy due to issues such as light distribution and chemical diffusion causing material to cure beyond intended pixel boundaries, thereby reducing resolution. The square shape of pixels can lead to a stair-stepping effect when printing circular or angular features, resulting in unexpected artifacts in the final printed geometry.

Various efforts have been made to enhance the accuracy of DLP printing, but many methods rely on trial-and-error approaches, necessitating extensive exploration of design parameters. g-DLP printing represents a recent advancement that leverages grayscale pixels to achieve smoother surface finishes. Utilizing the grayscale capabilities of pixels to address distortion shows promise for

improving printing quality, although determining the optimal grayscale values and arrangement remains a challenge. Physical modeling methods have been employed to simulate the correlation between curing processes and pixel grayscale, aiming to enhance the understanding of printing errors and minimize their impact by determining optimized grayscale pixels. While these simulation techniques offer a relatively high level of accuracy, they come with substantial experimental costs.

Therefore, this study proposed an efficient ML approach to determine the grayscale pixel intensities of projected images, with the goal of reducing experimental costs while ensuring precision. A ML model based on U-Net is employed to explore the correlation between printing edges and grayscale pixels. However, training a ML model necessitates a specific amount of experimental data, for which regularized chessboard data has been introduced as the training dataset. To streamline the entire data collection process, automated data correction techniques and a CGAN-based data augmentation method are integrated. Ultimately, this data-driven methodology is utilized to refine angular and circular edges by precisely controlling grayscale at the pixel level, thereby significantly enhancing the quality of 3D printing, particularly in terms of fine features and surface finishes.

Chapter 3 Research methodology

The objective of this thesis is to develop a ML model capable of comprehending the relationship between different combinations of grayscale pixels and their corresponding cured pattern profile (as shown in Figure 3-1a and b). This model can then be utilized to optimize the printed pattern profile by strategically distributing grayscale levels of pixels. To achieve this, a series of chessboard projection patterns were selected, as they offer the widest range of adjoining pixel grayscale level combinations. The resulting printed samples were washed and their appearance captured using an optical microscope, followed by a semi-automated high-throughput data refinement process (as illustrated in Figure 3-1c). These samples were then paired with the projection data to create a database for training the model. To expand the database size, a trained conditional Generative Adversarial Network (CGAN) [58, 76, 77] was introduced to generate artificial data (as depicted in Figure 3-1d), resulting in a database of 1200 chessboard data pairs. The final U-Net-based neural network was trained using a combination of experimental and artificial data, enabling us to determine the necessary projections for achieving desired features in manufactured objects (as shown in Figure 3-1e) [50, 78-80].

This approach was tested by challenging the model to replicate intricate structures and devices, including filters and microfluidic channels, and evaluating the level of improvement achievable via machine learning. The results demonstrated that the approach can enhance shape accuracy for both 2D patterns (as shown in Figure 3-1f) and 3D devices.

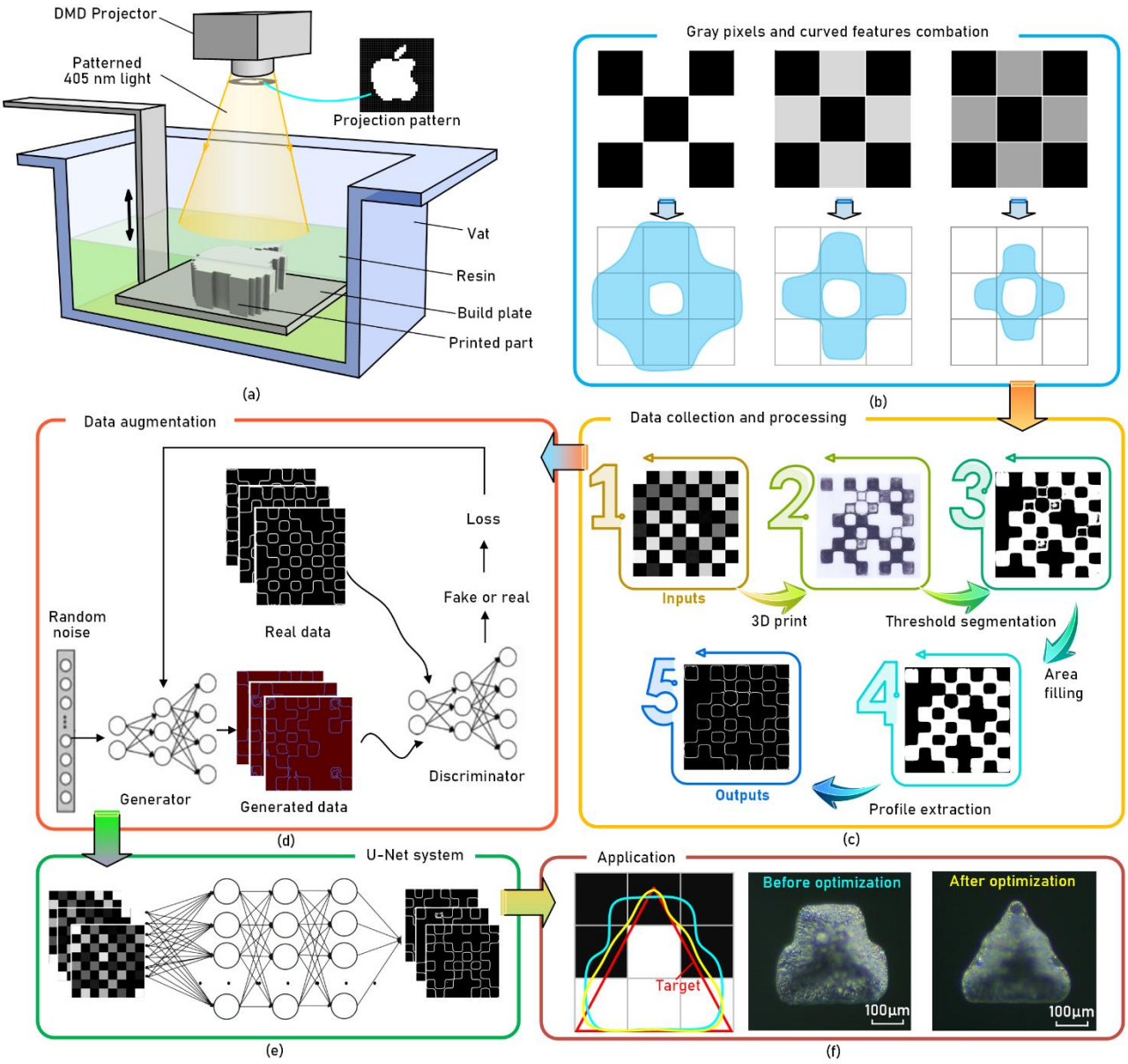


Figure 3-1 (a) Conceptual illustration of a lab-built vat photopolymerization printing apparatus utilizing a UV projector based on a digital micromirror device (DMD). (b) Schematic demonstrating how different combinations of grayscale pixels in the projection pattern can result in varying light intensities and solidification profiles. (c) Development of training data acquisition and processing steps based on the observations in (b) to gather a dataset for training a model to predict the relationship between the projection pattern and solidification profile. (d) Utilization of traditional and CGAN-based data augmentation processes to expand the dataset for training. (e) Establishment of the correlation between the projection pattern and curing pattern using the U-Net system. (f) Evaluation of the trained model's final performance.

3.1 Data collection

3.1.1 3D printer development and sample post-processing

A lab-built vat photopolymerization printer, illustrated in Figure 3-2 a, was equipped with a UV projector (GVINDA, model PDC05-70, with a printing resolution of 66-75 µm and optical distortion of 0.10%) emitting light at a wavelength of 405 nm and utilizing a 0.65 inch 6500 DMD chip. The chip's resolution was set at 1920x1080 pixels. The printing process involved submerging the build stage in a resin vat. Following the completion of each layer, the build stage was lowered, enabling a fresh layer of liquid resin to cover the surface. The UV light pattern was then cast onto the release film, creating a cured pattern on the build platform (see Figure 3-2 b). The UV projector maintained a steady intensity of 25.6 mW/cm^2 with an exposure duration of 3 seconds throughout the printing process.

The 3D printed samples were cleaned in an ultrasonic cleaning device and air-dried. These samples were then imaged under an optical microscope (KEYENCE, VHX-1000), ensuring that the images captured were oriented correctly. The original size of the projected image was 32x32 pixels, whereas the images captured by the optical microscope were 2740x2740 pixels. For comparative analysis, the projected image was first resized to 512x512 pixels, and the microscope image was similarly scaled down to 512x512 pixels for subsequent alignment.

For this study, three different commercial resins were utilized. The experiment began with a transparent UV resin (Resin ‘A’, Anycubic Classic), and was extended to include two additional UV resins (Resin ‘B’, a white Creality UV-sensitive resin, and Resin ‘C’, a grey Anycubic ABS-like Resin+). These resins were chosen for their distinct material properties, colors, and levels of translucency, showcasing the model's capability to adapt to various resin types.

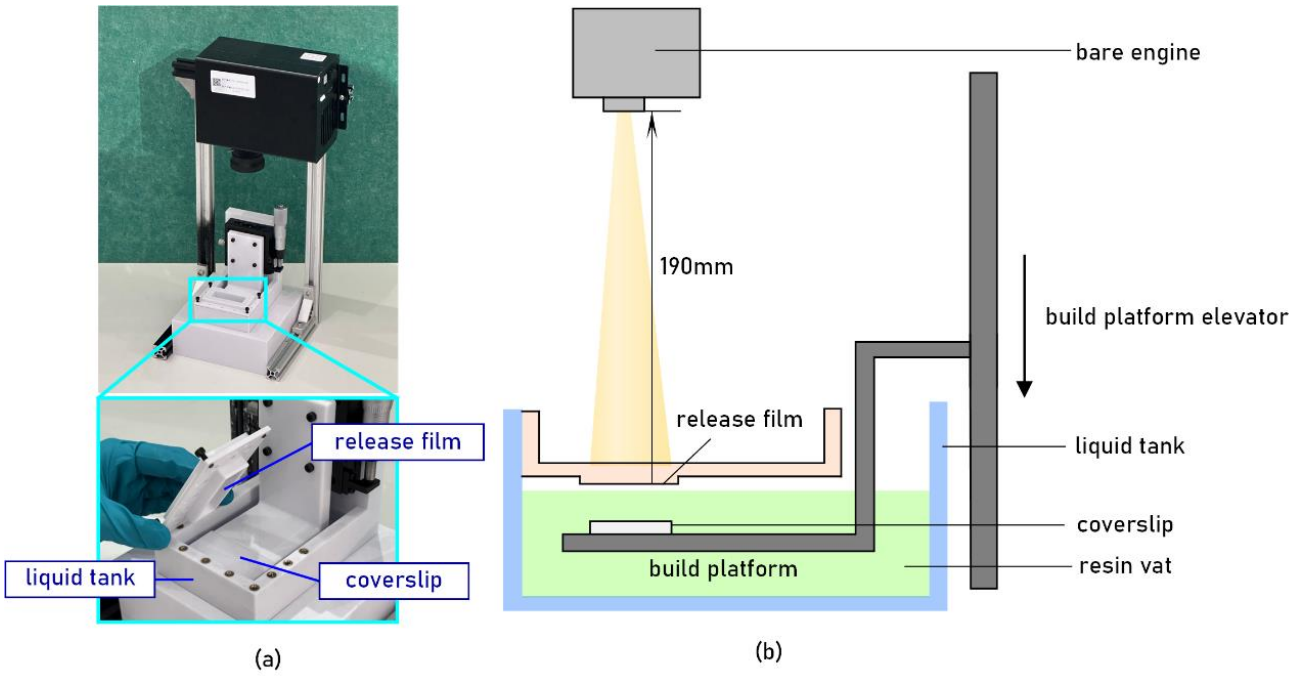


Figure 3-2 A schematic representation of the lab-built vat photopolymerization printer: (a) shows the overall system and the printing area, while (b) illustrates the configuration of the printer.

3.1.2 Projection strategy

The patterns projected followed a chessboard approach, where one set of diagonals consisted of fixed black pixels (0), while the other set comprised pixels with grayscale values randomly ranging from 0 to 255. This chessboard pattern was employed to optimize the distribution of grayscale values, creating a balanced mix of light and dark regions across the print surface. The structured nature of the chessboard pattern effectively mimics and balances light distribution during the curing process, making it a suitable approximation for illuminating and curing intricate geometrical shapes. By studying the curing effects of these fundamental patterns using ML, the knowledge gained can be applied to more complex patterns and structures, enhancing the efficiency of the printing process.

Each 2D 'chessboard pixel' was defined as a 4x4 array of 'projector pixels', forming an 8x8 chessboard upon projection. This approach was chosen for its simplicity in fabrication and analysis, although it can easily be scaled down to 2x2 projector pixels or expanded as needed. Throughout this paper, the term 'pixel' will refer to a 'chessboard pixel'. A total of twenty chessboard patterns were generated in a single batch, with a grand total of 300 patterns produced.

3.2 Data alignment

After 3D printed sample was captured by optical microscope, the captured image was converted into a binary file to simplify the data and emphasize important features, paving the way for more in-depth analysis and interpretation. Pixels were categorized as 0 or 1, typically representing black and white, respectively. In this study, a threshold of 127 was chosen, where pixel values above 127 were considered white and those below 127 were considered black.

Aligning the binary image with the projection pattern enabled the establishment of a mapping relationship between them. Image registration was ensured by fixing the top-left and bottom-right corner pixels to a value of 255 in all projections (see Figure 3-3 a).

After rough alignment using calibration points, further alignment is mainly achieved by calculating the intersection area of white pixels between the projected image and the cured image. The maximum value of the intersection area achieves the best match between the projected image and the cured image. However, since the chessboard we project is randomly set up, it cannot guarantee a sufficient number of pure white pixel blocks to help us achieve this best match. According to experimental tests, we found that small blocks with grayscale values greater than 200 can always be projected onto solidified blocks, meaning they will always generate corresponding white pixel blocks on the cured image. Based on this, we adjusted the algorithm to set small blocks with grayscale pixel values exceeding 200 to 255 during the alignment process, in order to expand the basis for calculating the intersection region and enhance the entire alignment process.

The intersection area of all white pixels in the projection pattern (A) and the binary image (B) was calculated. All white pixels in both images were set to 1, while all other pixels were set to 0. The white intersection area was determined by the formula [81]:

$$\mathrm{C}(\mathrm{x},\mathrm{y}) = \mathrm{A}(\mathrm{x},\mathrm{y}) \cap \mathrm{B}(\mathrm{x},\mathrm{y}) \tag{1}$$

where $\cap$ represents the AND logic operation, resulting in a white pixel (value 1) only if both A and B have white pixels at the corresponding position.

Alignment between A and B was achieved by maximizing the overlapping area (max_area). This involved two steps: initially, a rough alignment was performed by iteratively applying various

transformations to the binary image (B), including horizontal and vertical shifts (up to 20%) and rotations (up to 20°) with a step size of 2; subsequently, the variations were reduced to horizontal and vertical shifts (up to 5%) and rotations (up to 5°) with a step size of 1 [82]:

$$B' = B \times S \times R \times T \tag{2}$$

where B' is the transformed binary image, S is the scaling matrix, R is the rotation matrix, and T is the translation matrix.

The intersection area of A and B' was calculated, and if the area exceeded max_area, B and max_area was updated, with B' replacing B. The loop continued until the area no longer increased or reached the maximum transformation range.

Upon loop completion, the maximum intersection area (max_area) was determined, and the corresponding B represented the image with the greatest overlap of white areas between the two images, signifying successful matching. Any excess edges were then cropped based on the projection image (see Figure 3-3 b).

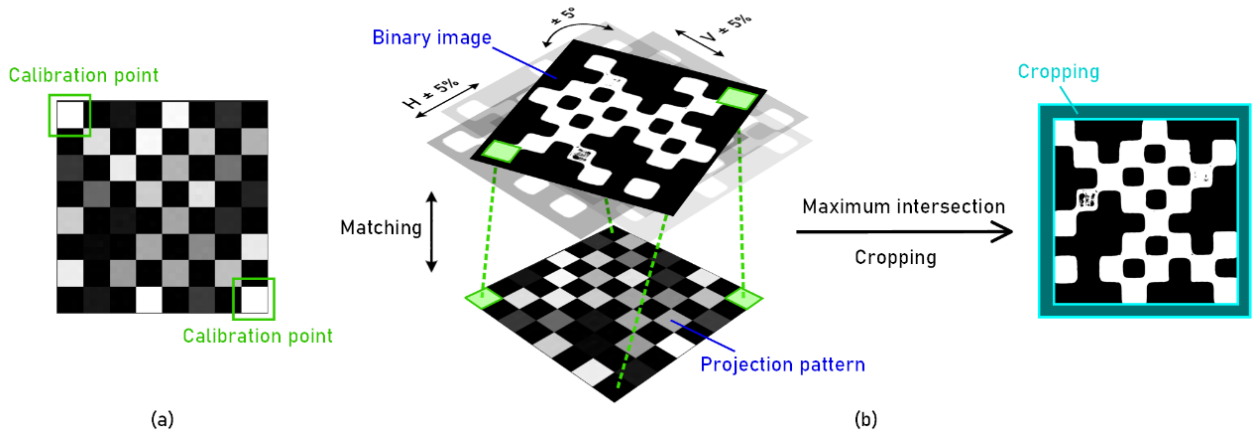


Figure 3-3 Calibration process of converting binary images into datasets featuring the projection pattern: (a) the chessboard layout with calibration points, while (b) illustrates the automated adjustment of the binary image to align with the projection pattern.

3.3 Data refining

Following the generation of noisy data from the manufacturing/data collection process, the subsequent step involved refining and curating the data. The majority of these imperfections

manifest as small black areas in the binary image (See Figure 3-4). To address this, we calculated the pixel area of these black regions and established a threshold to fill regions with a contiguous black pixel area smaller than 400 with white directly. The threshold value of 400 was determined through a series of experiments.

A portion of the data was removed by identifying structures that were likely smaller than what could be realistically achieved with the projection system, and subsequently eliminating them based on size thresholds. However, during the experiments we found that this approach runs the risk of inadvertently filling in correct features, particularly in cases where the correct structure is relatively small (as illustrated in Figure 3-4 d), resulting in some erroneous fillings (See Figure 3-4-B).

In the realm of image processing and computer vision, gradients play a fundamental role in essential tasks such as edge detection, feature extraction, and object detection. In binary images, where there are only two grayscale levels, the computation of gradient values and directions is relatively straightforward. The visual representation of grayscale variations through gradients provides an intuitive means of analysis.

Upon examination of these binary images (See Figure 3-4 c), we can notice that correct features are typically surrounded by large continuous white regions in both vertical and horizontal directions, resulting in a lack of gradient variations within a specific range. Conversely, incorrect features do not exhibit this characteristic in both directions, leading to dual occurrences of gradient shifts within the same distance range. Building upon this observation, refinements have been introduced to existing methods to effectively leverage the presence or absence of gradient alterations along the vertical and horizontal directions, thereby facilitating a more precise differentiation between accurate and inaccurate features.

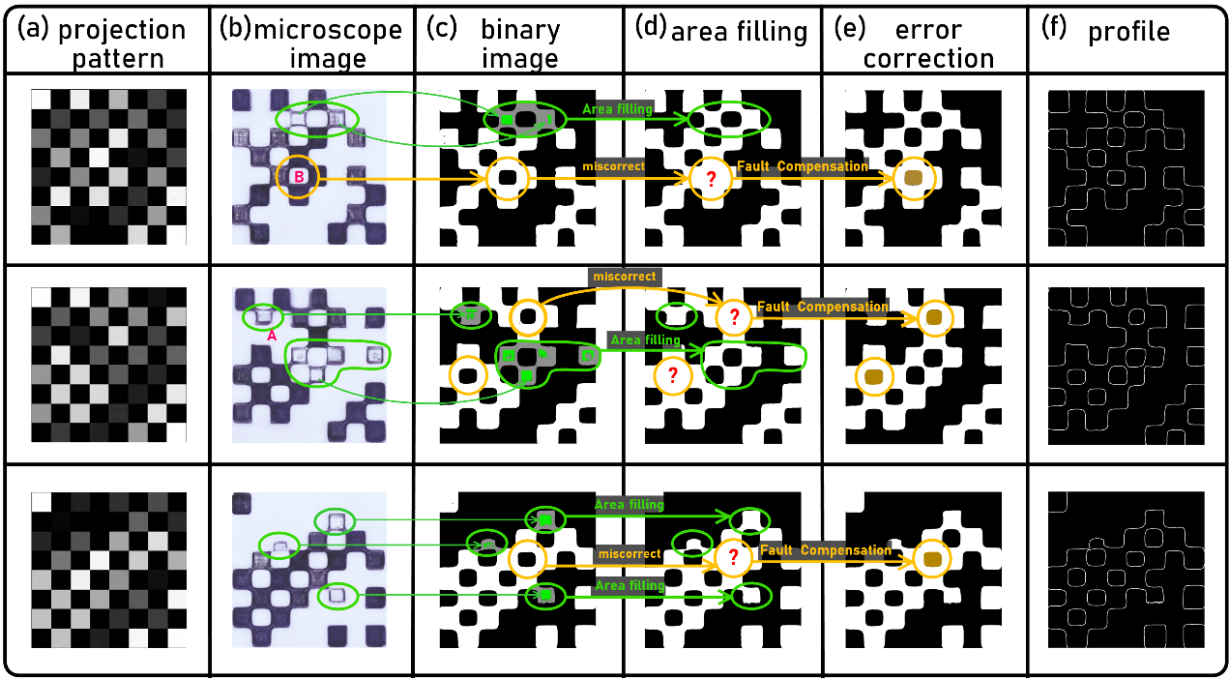


Figure 3-4 Optical microscope image processing steps: (a) display of a grayscale chessboard pattern, (b) optical microscope image of a grayscale chessboard obtained through Vat photopolymerization 3D printing, (c) segmentation of optical microscope images using a threshold, (d) binary image post area filling, (e) binary image following error correction, and (f) extraction of profiles from binary images.

Encircling the correct features are extensive regions of white pixels in both horizontal and vertical orientations (See Figure 3-6), with the gray region at the center (Figure 3-6 a) indicating the erroneously filled correct feature. In binary images, pixels are represented as either white (1) or black (0). To assess whether the central area is enclosed by purely white projections, we introduced gradient vectors. The gradient G(i,j) and direction θ(x,y) were computed as follows:

$$G(i,j) = \sqrt{(G_x^2(i,j) + G_y^2(i,j)} \tag{3}$$

$$\theta(x,y) = \arctan\left(\frac{G_y(x,y)}{G_x(x,y)}\right) \tag{4}$$

where i, j are the pixel indices, G_x and G_y represent the horizontal and vertical directions.

Therefore, the gradient vector value was determined using the approach illustrated in Figure 3-5. In this method, for the target vector direction, if there are two consecutive pixels along the vector direction that differ, the gradient of that vector is assigned as 1; otherwise, it is set to 0.

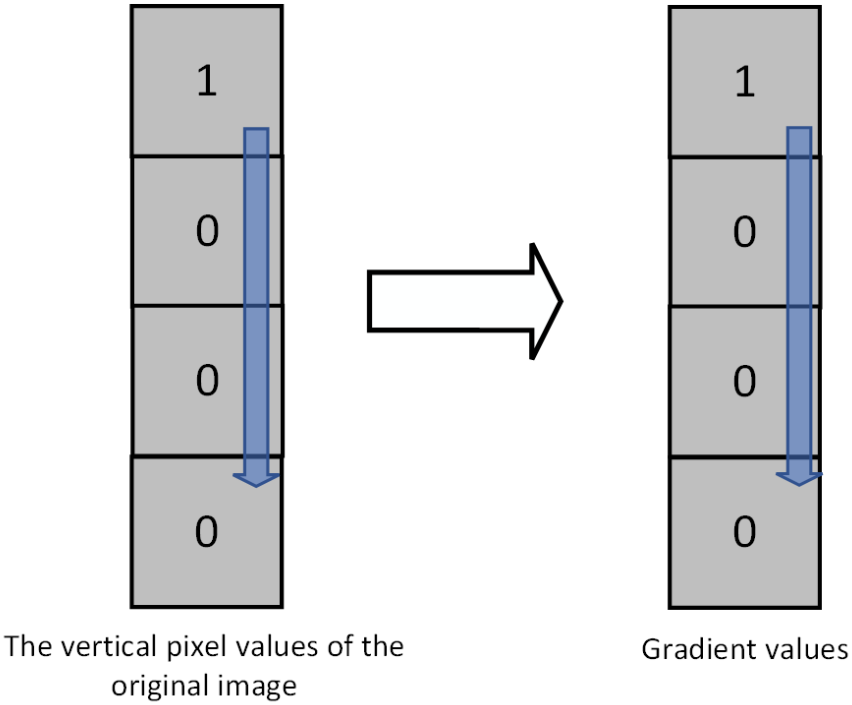
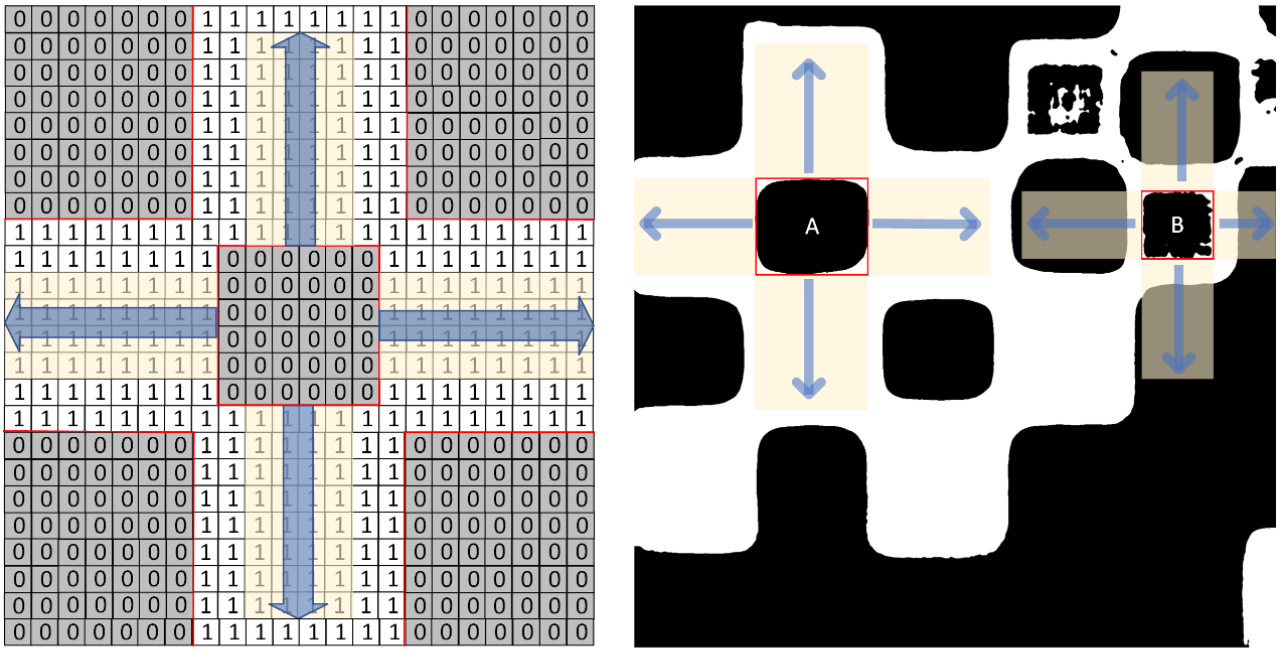


Figure 3-5 The computation of gradients for a binary image, using the vertical direction as a specific example.

In Figure 3-6 a, four distinct vector directions (indicated by blue arrows) were utilized for evaluation. For each vector direction, a 64*32 pixel rectangular area (highlighted in yellow) was selected to compute its gradient vector value. Each vector initiates from the border of the black region, thereby excluding the initial color transition between itself and the surroundings from consideration. If the area corresponds to the correct feature, the gradient vectors for all directions should register as zero (Figure 3-6 b-A); otherwise, the value will be 1 or higher. Consequently, this evaluation was incorporated into the filling algorithm, wherein if the gradient vector of the target region equated to 0, the black area (400 pixels or less) would not be filled. This methodology revealed that the black areas resulting from light scattering in the optical microscope (Figure 3-6 b-B) were predominantly allocated and filled accurately without compromising the correct features. Figure 3-4 e illustrates how this approach led to a reduction in erroneous data points.



(a) (b)

Figure 3-6 The computation of gradients for a binary image, with each small square in the diagram representing 8 pixels: (a) showcases the distribution of pixel values in the binary image; (b) demonstrates the gradient variations around correct and incorrect features. In this figure, blue arrows denote the direction of gradient calculation, while the yellow regions indicate the calculation range.

Moreover, utilizing the image directly for training the model would demand a significant amount of time. To enhance efficiency, image profiles can be extracted based on the gradient value and direction, where the gradient is always perpendicular to the edge where the shape appears. Profile extraction can be directly performed using the Canny parameters in OpenCV, a widely-used open-source computer vision and machine learning software library. OpenCV offers a broad spectrum of tools and functions for developing applications related to image and video processing, object detection, machine learning, and more. Within the Canny parameters, users can adjust the desired edge thickness and choose whether to fill the pixels inside the profiles. In this case, a thickness of 2 is set, the interior regions are not filled, and only the profiles are required. The resulting output is in png format (See Figure 3-4 f).

This approach effectively transformed a figure pattern into profiles, significantly reducing the computational load during the training process [83]. Consequently, the training speed increased by 5 times, resulting in a reduction of the processing time for each data pair from 890 ms to 178 ms (See Table 3-1).

Table 3-1 Comparison of model training time before and after profile extraction

Training data	epochs	Training dataset	Average processing time of a single image (ms)	Variance of processing time for a single image (ms^2)	Total training time of the model (h)
1200	150	Original dataset	890.5	0.75	40.046
		Dataset after profile extraction	178.5	0.67	8.041

3.4 Data augmentation

3.4.1 Traditional data augmentation

Initially, rotation was applied to the 300 sets of experimental data using traditional data augmentation techniques, which included projection images and their corresponding microscope images post threshold segmentation (See Figure 3-7 a). These techniques prove effective in enhancing the diversity of training data and boosting the model's generalization capabilities. However, the efficacy of traditional data augmentation is limited by data distribution, and the generated samples may not cover the entire data space. Consequently, once traditional data augmentation reaches a certain data volume, it ceases to provide additional benefits for model training and can even prolong the training time [84, 85]. Consequently, traditional data augmentation methods were employed to expand the training dataset from 300 to 600 data sets.

3.4.2 Conditional generative adversarial networks

To enrich pattern diversity and expand the available data, an integration of a Pix2Pix model, a variant of CGAN, was incorporated into the methodology for data augmentation. A trained Pix2Pix CGAN is designed to generate output images corresponding to conditioned input images [86]. Typically, it comprises a U-net based architecture and a discriminator. During training, the system is fed with paired input and output data to establish correlations between them. Consequently, when presented with an input data, the model can predict and generate an output data based on its training. While commonly utilized in AI drawing tasks, this approach can also be leveraged for data augmentation, particularly in scenarios involving paired conditioned patterns. This strategy offers an effective solution to the challenge of limited data and significantly enhances accuracy. Previous

studies by Lakmal [87] and Liu [88] have successfully employed Pix2Pix for data augmentation, demonstrating its proficiency in precise image-to-image mapping tasks, such as converting sketches into detailed images or transforming daytime photographs into nighttime scenes, thereby enriching dataset diversity. Moreover, Pix2Pix has shown the ability to produce realistic synthetic images, effectively expanding the dataset for model training.

The CGAN [59] has the capacity to generate data through a generator. In our context, the generator accepts a randomly generated pattern and a grayscale chessboard pattern as inputs to produce the cured pattern. The discriminator's role is to differentiate between generated and real data (See Figure 3-7 b).

The loss function utilized in the CGAN training process is as follows:

$$\min_{G}\max_{D} V'(D,G) = \mathrm{E}_{\mathrm{x\sim Pdata(x)}}[\log\ \mathrm{D(x|\,y)}] + \mathrm{E}_{\mathrm{z\sim Pz(z)}}\left[\log\left(1-\mathrm{D}\left(\mathrm{x},\mathrm{G(y|z)}\right)\right)\right] \quad (5)$$

Where y represents the specified condition, x denotes the actual data corresponding to condition y, z is the random noise input to the generator, G(y|z) represents the data produced by the generator based on condition y and noise z, and D(x|y) is the discriminator's probability assessment that the given data x under condition y is authentic.

To replicate the real-world scenario and generate a 3D printed pattern resembling it, we employed the Stochastic Gradient Descent [89] technique to optimize the generation parameters (outlined below):

$$\theta_{\mathrm{G}}{}' \leftarrow \theta_{\mathrm{G}} + \alpha\frac{1}{\mathrm{m}}\sum_{\mathrm{i=1}}^{\mathrm{m}}\frac{\partial\log\left(1-\mathrm{D}\left(\mathrm{G}(\mathrm{z_i}|\mathrm{y_i})\right)\right)}{\partial\theta_{\mathrm{G}}} \quad (6)$$

Here, θ_{G} represents the modifiable parameter in the generated model that governs the generator's operations and dictates how it generates samples from random noise. By manipulating the value of θ_{G}, we can alter attributes such as the quality, diversity, and authenticity of the samples produced by the generator. The parameter α signifies the learning rate, m denotes the sample count, D stands for the discriminator, and $\mathrm{z_i}$ and $\mathrm{y_i}$ individually denote a generator input and its corresponding condition label.

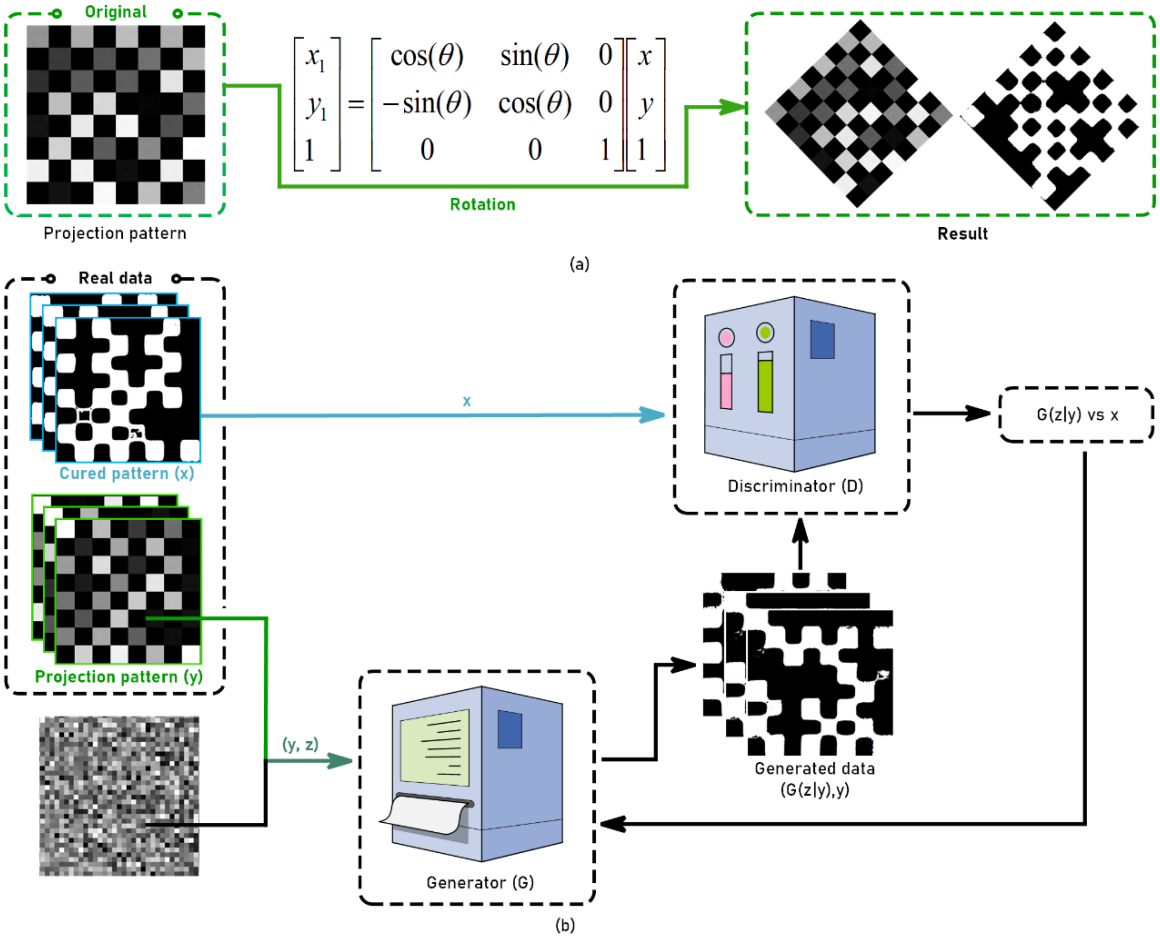


Figure 3-7 Data augmentation strategies employed in this study: (a) Traditional data augmentation techniques that broaden data by employing geometric transformations, and (b) Utilization of the Conditional Generative Adversarial Network (CGAN) for training and generating data to enhance the diversity of data patterns.

To ensure the accuracy of the data produced by the CGAN, a subset of the experimental data was chosen as validation data for testing purposes. The trained generator was provided with a projection pattern, and the resemblance between the resulting cured pattern generated by the generator based on this projection pattern and the actual experimental cured pattern was evaluated. The Dice coefficient similarity between the generated image and the experimental image was determined to be 89.7±2.8% (See Figure 3-8).

Subsequently, data processing operations were carried out on these data pairs, which included tasks like area filling and profile extraction as mentioned earlier. These processed data pairs were then compiled to create the final dataset for model training.

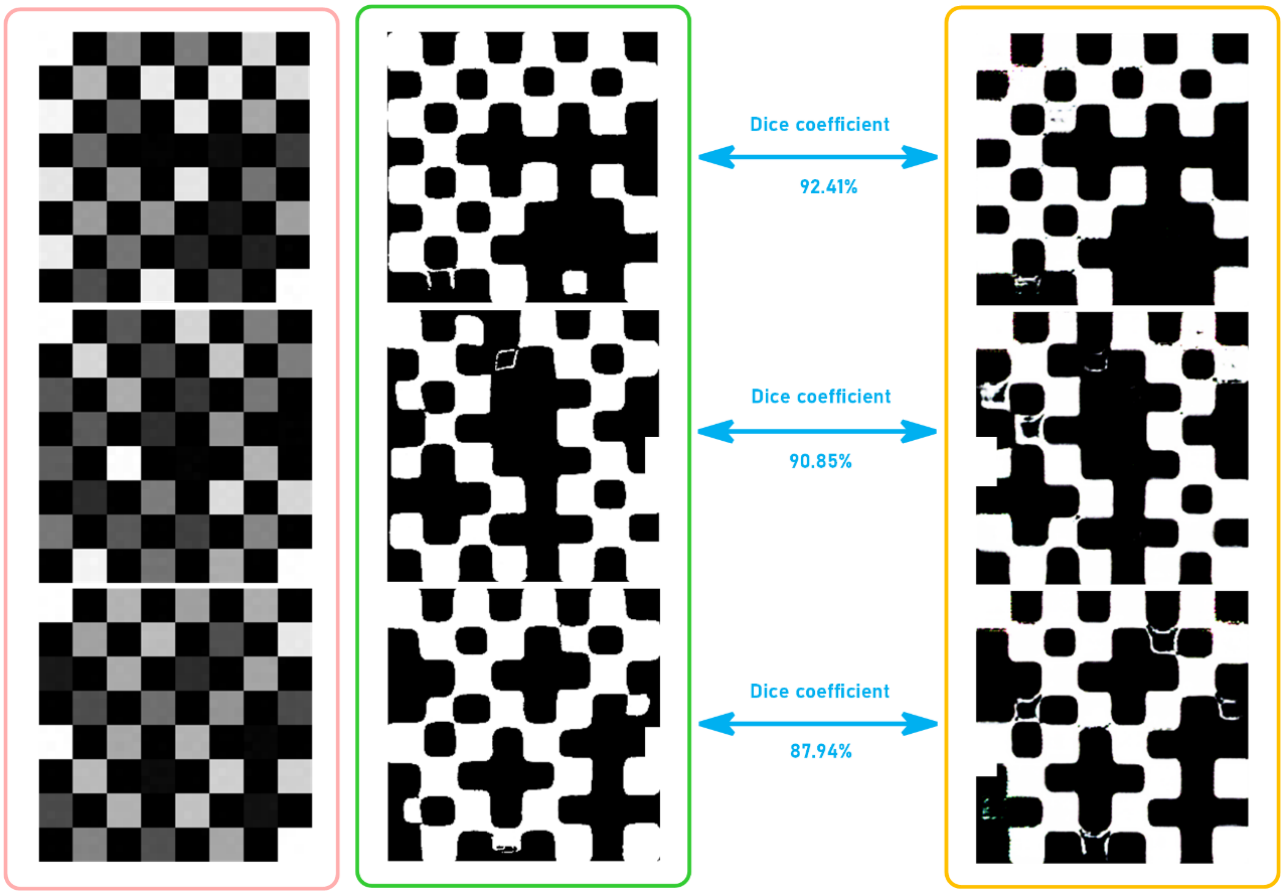


Figure 3-8 A comparison between images generated by CGAN and actual experimental images.

Through the implementation of the CGAN approach, the initial set of 600 datasets obtained from traditional data augmentation was effectively doubled to a total of 1200 datasets. It is important to highlight that, given the constraints of our current computing resources, training a dataset larger than 1200 is not feasible. However, in theory, with increased computational power, the dataset could be expanded further.

The data augmentation procedure was performed based on the pix2pix model of CGAN. The CGAN-pix2pix strategy was used (method provided by junyanz), reference link: https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix During the training process of the CGAN model, this study did not modify the loss function, but only adjusted the following hyperparameters:

1. Learning Rate: The learning rate is one of the most important hyperparameters, determining the magnitude of weight updates. For CGAN, the learning rate is typically set to a smaller value to ensure a stable training process. Here, a learning rate of 0.0001 was chosen.

2. Optimizer: The Adam optimizer is a common choice for training CGAN because it combines the

advantages of momentum and adaptive learning rates. For the Adam optimizer, the `beta1` parameter (momentum term) is set to 0.5, and the `beta2` parameter (the decay rate of the second moment estimate) is set to 0.999.

3. Batch Size: Batch size affects the model's memory usage and training stability. Here, a batch size of 64 was selected.

4. Epochs: The number of training epochs determines how many times the entire dataset is used to train the model. The number of epochs for CGAN training may need to be adjusted based on the size and complexity of the dataset, usually ranging from a few hundred to a few thousand. In this case, 300 epochs were used.

3.5 Model training

The U-Net architecture was initially tailored for biomedical image segmentation, known for its capacity to capture global information while preserving intricate features within images. This framework enhances the comprehension of input images, enabling precise pixel-level predictions. Consequently, it is well-suited for tasks requiring high precision in image translation, facilitating the accurate establishment of relationships between grayscale projection images and binary images. These correspond to the profile printed by varying grayscale pixels, ensuring accurate pixel-level predictions.

In the U-Net implementation, a hyperbolic tangent activation (tanh) function [90] was utilized and mapped to the 0-255 range to obtain the final grayscale image. This function, mathematically defined as:

$$\tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{7}$$

The function outputs values within the (-1, 1) range, aiding in zero-centering the data for efficient model learning. Centering data around 0 helps reduce bias and hasten convergence during training. The mapping from tanh output back to the 0-255 range can be achieved through the formula:

$$\text{output} = (\text{tanh\_output} + 1) * 127.5 \tag{8}$$

During training, a cross-entropy loss function[91] was employed to evaluate the disparity between

predicted and actual data:

$$L = -\sum_{i=1}^{N} y^{(i)} \log \hat{y}^{(i)} + \left(1 - y^{(i)}\right) \log\left(1 - \hat{y}^{(i)}\right) \quad (9)$$

Where i denotes the individual sample, with $y^{(i)}$ and $\hat{y}^{(i)}$ representing the actual and predicted labels of the model, respectively. N corresponds to the total number of samples analyzed.

This loss function measures the performance of classification models by comparing predicted probabilities to actual labels. A lower cross-entropy loss signifies improved model performance, as it captures the deviation between predicted probabilities and actual distributions effectively, facilitating faster model convergence.

And stochastic gradient descent [92] was utilized to update model parameters:

$$\theta' = \theta - \alpha \nabla_{\theta} L(y, \hat{y}) \quad (10)$$

where θ is the model parameter, α is the learning rate, $\nabla_{\theta} L(y, \hat{y})$ represents the gradient of the Loss function to the model parameters. In each iteration, a randomly selected sample set is used to compute the gradient, updating the model parameters until optimal parameters minimizing the gradient are attained.

In the training process of the U-Net model, followed the protocol from Ronneberger's paper, which includes code and public datasets. DOI: 10.1007/978-3-319-24574-4_28

The values of the key parameters are as follows:

Learning rate: The learning rate for training U-Net is typically chosen as 0.001 because this value has been found to work well for optimizing the model's parameters efficiently without causing instability or slow convergence during the training process.

Loss function: Cross-entropy loss function is used for training U-Net because it is well-suited for semantic segmentation tasks where pixel-wise classification is involved. Cross-entropy loss helps to effectively penalize classification errors and provides a good measure of the discrepancy between the predicted segmentation map and the ground truth, enabling the network to learn and optimize its parameters for accurate pixel-wise predictions.

Optimizer: The Adam optimizer is chosen for training U-Net due to its ability to efficiently update

the model's parameters by adapting the learning rates for each parameter individually. This adaptive learning rate method helps in speeding up convergence and handling sparse gradients, which is beneficial for training deep neural networks like U-Net effectively.

The Adam optimizer was used in the framework of tensorflow[93] to train the network in small batches of stochastic gradient descent[89]. Due to the memory limit (4 GB), the batch size is set to 16. The entire training process uses the Quadro P620 graphics processing Unit (GPU). The model architecture replicates the architecture of U-net with some adaptations[50]. The network consists of 14 modules. The first seven blocks each have a convolution layer with stride 2 to up sampling features. The convolution layer is followed by a batch normalization layer and a ReLU function[94]. The other seven blocks each have a stride-2 deconvolution layer to up sampling the features. With the exception of the last layer, which uses the hyperbolic tangent function (Tanh)[90], all the other layers use ReLU, which is to ensure that the final output layer is sorted by pixel and outputs grayscale images, rather than the single-channel output of the main neural network. After training, the neural network can calculate the output grayscale projection on a personal computer in just a few seconds.

Figure 3-9 illustrates the model training process, where the training dataset was split into training (80% of the data: 960 pairs) and validation (20% of the data: 240 pairs) groups. The training group was utilized to train the ML model, establishing the correlation between projected pattern grayscale images and 3D printed profile data. The validation group tested the model's performance on unseen data to adjust training parameters.

Following this, 3D printed profiles from the validation group were input into the trained model to generate corresponding projection patterns. By comparing the generated and original projection patterns, deviations were measured to refine the model. Mean Squared Error (MSE) was employed to evaluate the discrepancy between computed and original projected patterns during the training process.

It is important to emphasize that MSE serves as a tool for monitoring the training process and does not serve as a substitute for the loss function. The MSE calculation is defined as:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \widehat{Y}_i\right)^2 \tag{11}$$

Here, n denotes the sample size, Y_i represents the actual data, and $\hat{Y}_i$ signifies the predicted data generated by the trained model.

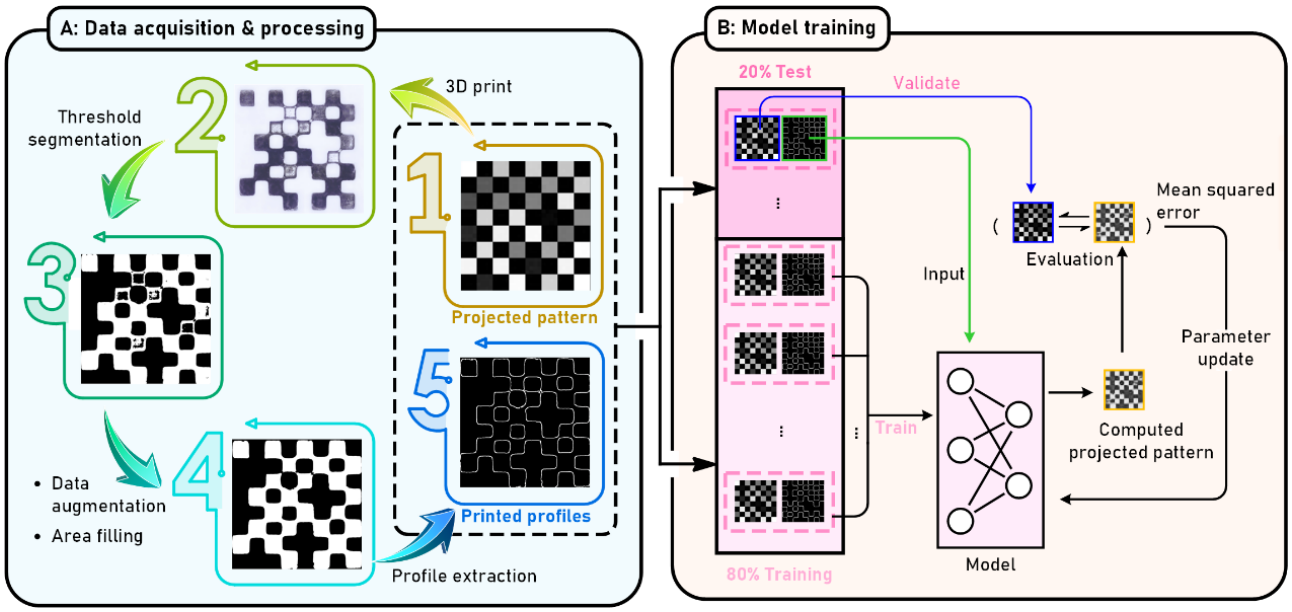


Figure 3-9 Process of training the model: (A) involves acquiring and processing chessboard data to derive the ultimate training dataset, while (B) entails partitioning the datasets, conducting model training, and validation.

The process of model training involves iterative adjustments to parameters, enabling the model to learn from the training dataset and exhibit strong performance within that dataset. However, relying solely on the reduction of MSE within the training data may not accurately reflect the model's overall performance. The primary focus lies in understanding the model's behavior when presented with new, unseen samples, aiming to minimize errors on unfamiliar data. Hence, the division of the dataset into training and validation sets plays a pivotal role in assessing model performance, fine-tuning hyperparameters, and guarding against overfitting. By training the model on the training set and evaluating it on the validation set comprising unseen data, a comprehensive assessment of the model's generalization capabilities is achieved.

Figure 3-10 illustrates that the MSE values for the training and validation datasets exhibit proximity and diminish as the number of parameter updates increases. This trend indicates the efficacy of the training process and the model's ability to generalize well beyond the training data. However, during the plotting process, a misconfiguration of a parameter led to the training curve being fitted to the validation data, resulting in a smooth curve that lacked the normal fluctuations seen in the training data's curve (See Figure 3-10 left). Upon rectifying this error, as shown in the right plot of Figure 3-

10, the training curve was corrected. A comparison between the curves before and after correction reveals slight fluctuations in the training curve. These fluctuations are attributed to the random sampling of the training data in each training epoch, leading to minor variations in data distribution and are considered a normal phenomenon.

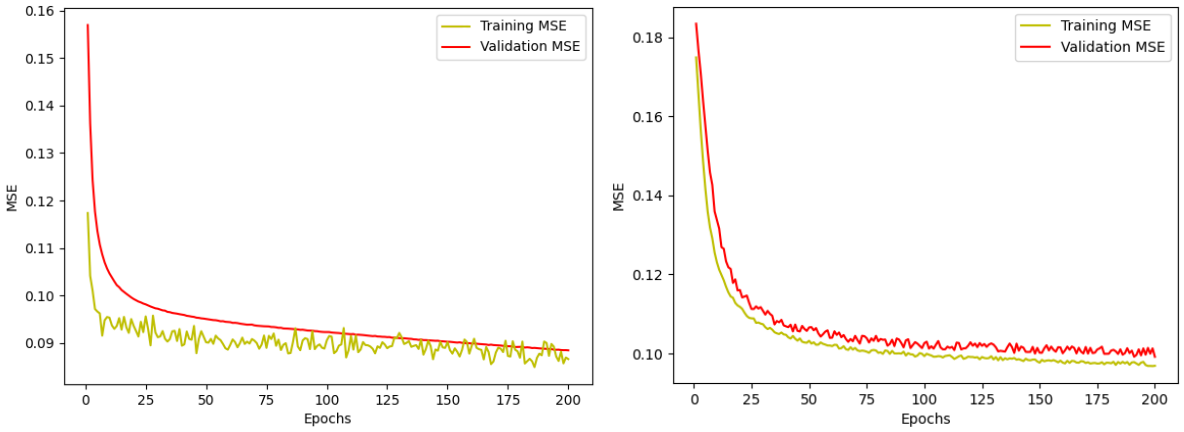


Figure 3-10 The evolution of Mean Squared Error (MSE) throughout the entire model training process before and after error correction.

3.6 Model evaluation

To assess the model's performance and quantify its accuracy, two coefficients were introduced to provide a quantitative evaluation of the shape accuracy of the 3D printed structure.

Dice coefficient

The Dice coefficient [95] is commonly utilized in image processing to evaluate the level of overlap and similarity between two images, making it a valuable metric for quantifying the performance of ML processes.

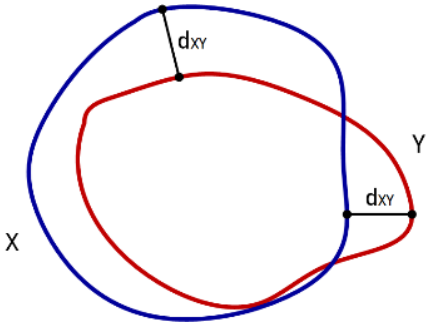
$$\mathrm{Dice}(X, Y) = \frac{2 \times |X \cap Y|}{|X| + |Y|} \tag{12}$$

where $|X|$ represents the cardinality of set X, $|Y|$ represents the cardinality of set Y, and $|X \cap Y|$ represents the cardinality of the intersection of X and Y. This coefficient ranges from 0 to 1, with a value closer to 1 indicating a higher degree of similarity between the two images.

Hausdorff distance

Furthermore, the Hausdorff distance [96], another metric introduced, focuses on precise matching

and is commonly used in identifying positional errors and other feature extraction techniques.



$$\text{Hausdorff}\,(X, Y) = \max[\sup_{x\epsilon X} \inf_{y\epsilon Y} d(x, y), \sup_{y\epsilon Y} \inf_{x\epsilon X} d(x, y)] \quad (13)$$

Here, d (x, y) represents the distance between x and y, with 'sup' indicating the upper limit and 'inf' representing the lower limit.

The calculation of these coefficients is facilitated through the utilization of an open-source Python library. Specifically, scikit-learn and SciPy are two Python libraries that cater to scientific and technical computing, offering a wide array of mathematical algorithms and functionalities supporting various operations such as linear algebra, optimization, integration, interpolation, special functions, fast Fourier transforms, signal processing, and image processing.

Within the SciPy library, a function named directed_hausdorff, responsible for computing the Hausdorff distance, can be found in the scipy.spatial.distance module, allowing for direct computation. While the implementation of the Dice coefficient can be achieved using Scikit-learn, it is worth noting that Scikit-learn does not provide a direct function for computing the Dice coefficient. However, this coefficient can be easily calculated using simple functions or by utilizing the jaccard_score function available in the sklearn.metrics module as an approximation, given the direct mathematical relationship between the Dice coefficient and the Jaccard index:

1. The Python code for implementing it with a simple function is as follows:

```
import numpy as np

def dice_coefficient(y_true, y_pred):
    intersection = np.sum(y_true * y_pred)
    return (2. * intersection) / (np.sum(y_true) + np.sum(y_pred))
```

2. The Dice coefficient can be calculated using the Jaccard index, with the specific formula as follows:

$$\mathrm{Dice}(\mathrm{A},\mathrm{B}) = 2 * \mathrm{Jaccard}(\mathrm{A},\mathrm{B})/\big(1 + \mathrm{Jaccard}(\mathrm{A},\mathrm{B})\big) \tag{14}$$

Where A and B represent two sets, respectively.

Nevertheless, it is essential to highlight that prior to initiating the computation module, the output from the printing process necessitates data processing for automatic calibration, as detailed in Section 3.2.

While both the Dice coefficient and Hausdorff distance play pivotal roles in assessing the accuracy of image shapes, they each focus on distinct aspects of segmentation precision. The Dice coefficient places emphasis on the overall similarity between the target and actual outcomes, acting as a dependable measure of general accuracy. Conversely, the Hausdorff distance draws attention to the maximum boundary differential, providing valuable insights into the precision of key points. By integrating these metrics, a comprehensive evaluation is achieved, ensuring not only accurate overall results but also finely detailed delineations.

Chapter 4 Model Performance

4.1 The effectiveness of data augmentation

Figure 4-1 depicts the progression of 3D printed sample quality with an increasing number of training datasets. It is apparent that when 100 datasets were utilized for model training, the printed snowflake exhibited numerous errors, including the merging of detailed structures such as branches. However, as the quantity of training data increased, the quality of the printed snowflake notably improved, displaying enhanced details and accuracy. The snowflake's branches became more distinct, and the overall pattern appeared more symmetrical. To provide a more precise description of this progression, the evolution of snowflake fidelity with the number of datasets was measured using the Dice coefficient and Hausdorff distance (See Section 2.7). The Hausdorff distance decreased significantly with the expansion of model training data, while the Dice coefficient exhibited a gradual increase (See Figure 4-1 g-h), indicating a closer resemblance of the 3D printed snowflake pattern to the target. Significance testing also confirmed the statistical significance of this improvement.

From Figure 4-1, it is evident that the surface of the printed snowflake samples is not smooth. This is due to the fact that the samples initially grow on the release film, with one side in contact with the smooth surface of the film while the other side typically remains non-smooth. Ideally, during normal printing, the samples should naturally detach from the release film as the build platform moves downward. However, inadequate curing depth causes the samples to remain adhered to the release film instead of detaching. To preserve the integrity of the samples and prevent damage, we chose to observe the samples directly on the release film without removing them, resulting in the observed non-smooth surface. The adhesion issue may also be attributed to the material of our printing platform. We envision that constructing this machine with metal components in the future could potentially resolve this issue.

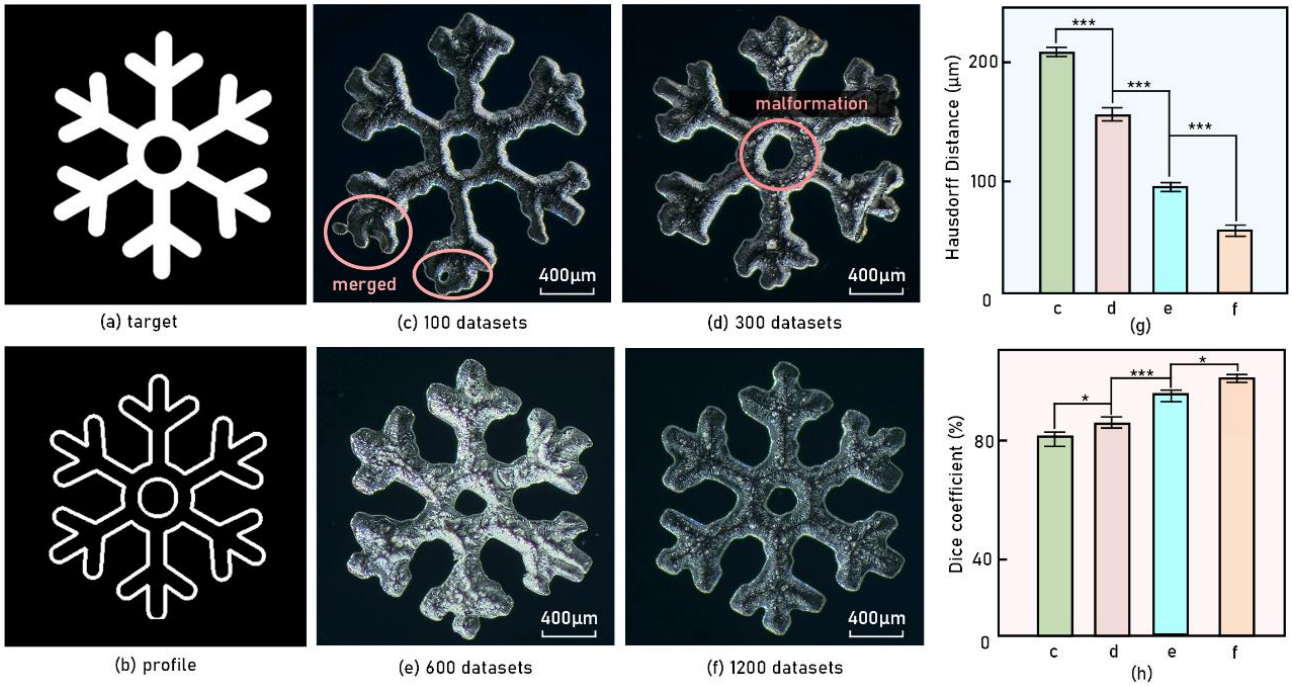


Figure 4-1 The impact of data augmentation techniques: (a) shows the intended target while (b) displays the profiles of the target. The 3D-printed snowflakes in (c) were produced by a model trained on 100 datasets. Subsequent images (d), (e), and (f) depict snowflakes printed by models trained on 300, 600, and 1200 sets of training data, respectively. The evolution of snowflake fidelity with the number of datasets is quantified in (g) and (h) using the Hausdorff distance and Dice coefficient. Statistical significance was evaluated using a one-way Anova followed by Tukey's post hoc test (indicates $p < 0.05$, ** indicates $p < 0.01$, and *** indicates $p < 0.001$ for the comparisons between the datasets).*

4.2 Model performance of Geometric patterns

To assess the model's performance following the proposed training approach, two pixel-scale designs, a circle, and a triangle, were used as challenges (See Figure 4-2). As depicted in Figure 4-2 A1 and A3, the 3D printing resolution limitations result in a 3-pixel diameter circle being sliced into a simple cross, failing to achieve the desired design. However, optimization by the trained model enables the attainment of such small curvatures through the appropriate combination of grayscale pixels.

In Figure 4-2 A3 and B3, the outer profiles of the designs and the actual 3D printed sample profiles were extracted from the projection patterns pre and post-optimization. It is evident that the maximum deviation distance before optimization (H1) is significantly larger than that observed with the optimized pattern (H2).

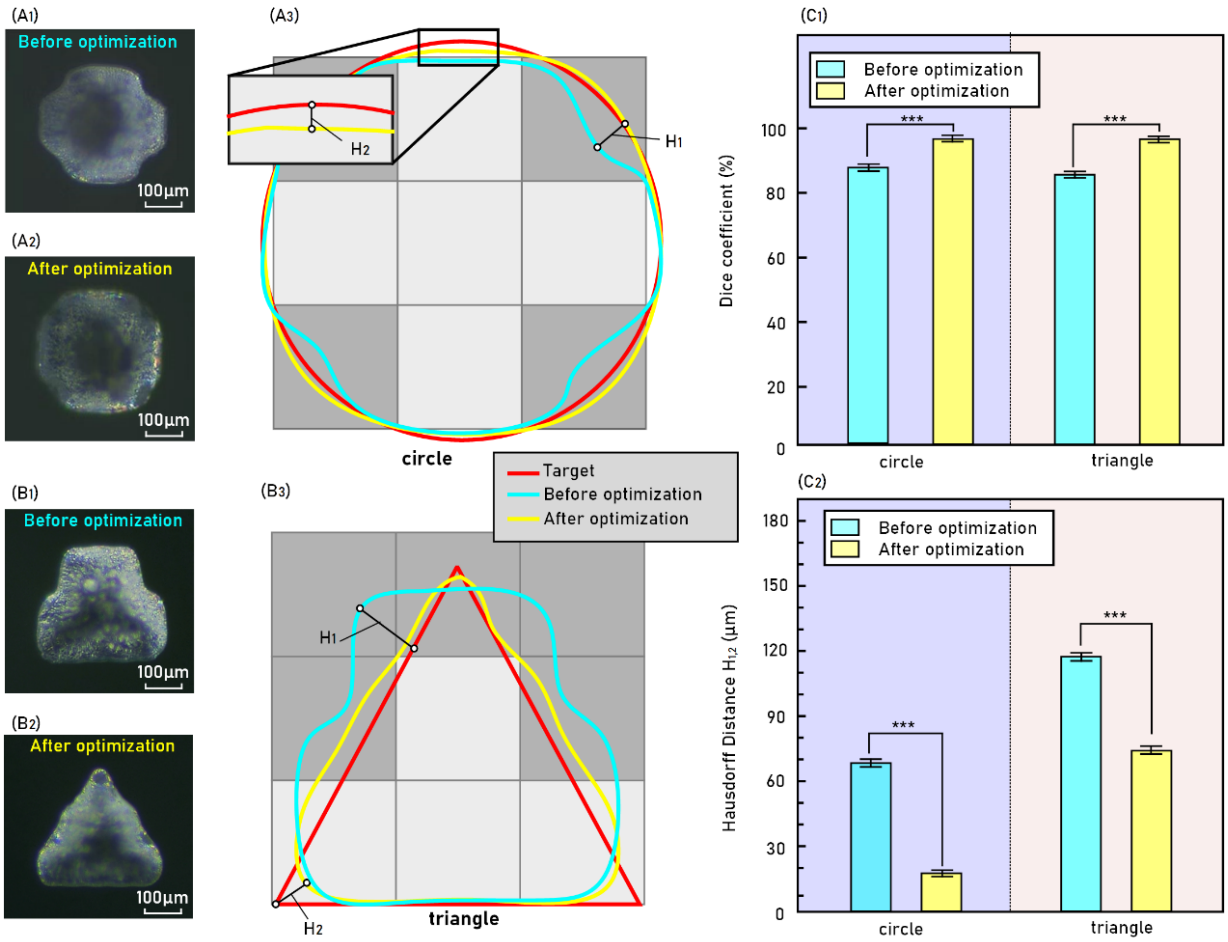


Figure 4-2 Two pixel-scale designs that were 3D printed to evaluate the impact of the model: (A1) shows a 3D printed circle with a 3-pixel diameter before optimization, while (A2) displays a 3D printed circle with a 3-pixel diameter after optimization using the proposed method; (A3) illustrates the overlap of the 3D printed circle and the target circle. Similarly, (B1) depicts a 3D printed triangle with 3-pixel-long sides before optimization, and (B2) shows a 3D printed triangle with 3-pixel-long sides after optimization; (B3) presents the overlap of the 3D printed triangle and the target triangle. The Dice coefficient (C1) and Hausdorff distance (C2) were used to quantify the results. Statistical significance was determined using a one-way Anova with post hoc Tukey's test (indicates $p < 0.05$, ** indicates $p < 0.01$, and *** indicates $p < 0.001$ for the comparisons between the two).*

The utilization of both the Dice coefficient and Hausdorff distance facilitated a quantitative comparison of the improvements in shape accuracy. These metrics provided a comprehensive assessment of enhancement from distinct viewpoints. Post model optimization, the Dice coefficients of both simple patterns increased by approximately 10% (as illustrated in Figure 4-2), indicating an overall improvement. However, concerning the maximum boundary deviation measured by the Hausdorff distance, the enhancement in accuracy was more pronounced for the circular structure compared to the triangular design. Specifically, the design accuracy improved by 80% for the triangle

and approximately 300% for the circle (as shown in Figure 4-2 C2). It was observed that the primary deviation stemmed from the corners of the triangle structure. Shapes with sharp corners are particularly sensitive to minor deviations, a characteristic better captured by the Hausdorff distance. The substantial reduction in both parameters implies a significantly enhanced fidelity of the 3D printed shape compared to its previous state. Notably, this enhancement was accomplished without the need for any hardware upgrades.

4.3 Model performance of complex patterns

To further evaluate the optimization method's efficacy, the model was tested with intricate, small patterns (See Figure 4-3).

Observing Figure 4-3 a, it is evident that the butterfly's antennae (at location i) closely resembled the intended pattern when utilizing the model. Although neither rendition fully replicated the intricate hollow patterns in the butterfly wings, the optimized version displayed enhanced printing quality and captured more of the hollow structures compared to the non-optimized version (locations ii and iii).

Similar enhancements were observed in the bug and watermelon patterns, showcasing the heightened fidelity achievable with the trained model. In the bug pattern, the bug's 20 µm legs exhibited significant improvement in curvature and void depiction. In contrast, the bug printed using the non-optimized pattern tended to merge its legs, resulting in notably inferior detailing (locations iv and v).

Comparable improvements were also noticeable in the watermelon pattern, featuring both large (location vii) and small curvatures (location vi). The samples printed using the non-optimized patterns tended to represent these curvatures as right angles (location vii) or zig-zag paths (location vi), a flaw rectified by the trained model, eliminating step-like appearances and enhancing the reproduction of the original design's curvature.

As before, the Dice coefficient was employed to quantify the enhancement in the samples printed with and without optimization. Figure 4-3b illustrates an average improvement in shape accuracy of approximately 10-20%. The analysis of the Hausdorff distance for the intricate patterns revealed notable enhancements in fidelity for the butterfly and watermelon designs, with improvements of

approximately 45% and a substantial 105% enhancement for the insect designs. A decrease in the Hausdorff distance signifies a closer match in shape, size, and alignment between the printed features and the intended design, indicating an overall increase in fidelity post-optimization. While both simple and complex patterns displayed improvements of around 10-20% in terms of the Dice coefficient following optimization, a comparison based on the Hausdorff distance revealed a 4-15 times greater enhancement in simple patterns compared to complex ones. This disparity may stem from the varied features present in complex patterns, such as sharp corners, for which our current training data may lack sufficient optimization. Future research endeavors could focus on enriching the training data with intricate structures to further refine optimization accuracy.

The experimental outcomes utilizing the Hausdorff distance and Dice coefficient underscore the efficacy of grayscale processing in refining edges and closely approximating the target image, emphasizing the critical role of precise pixel-level control in mitigating edge inaccuracies [1]. The study showcases the effectiveness of the ML model while indicating that although both metrics are vital for evaluating image fidelity, they yield differing degrees of enhancement due to their emphasis on distinct aspects of segmentation accuracy (See Section 3.6).

During the process of printing complex patterns, initially, our designed intricate patterns were not centered in the projection image but rather randomly placed on the black background projection. However, we observed that varying the placement of the patterns resulted in poor uniformity and weak print repeatability. Through literature review, we learned that this inconsistency was due to uneven light intensity, stemming from the optical system. Since altering the light source of the optical system was not feasible, we optimized our experimental design by ensuring that each pattern to be projected is centrally positioned on the black background projection image every time to enhance experimental repeatability. Similarly, as mentioned in Section 4.1, the surface of the printed samples remains insufficiently smooth.

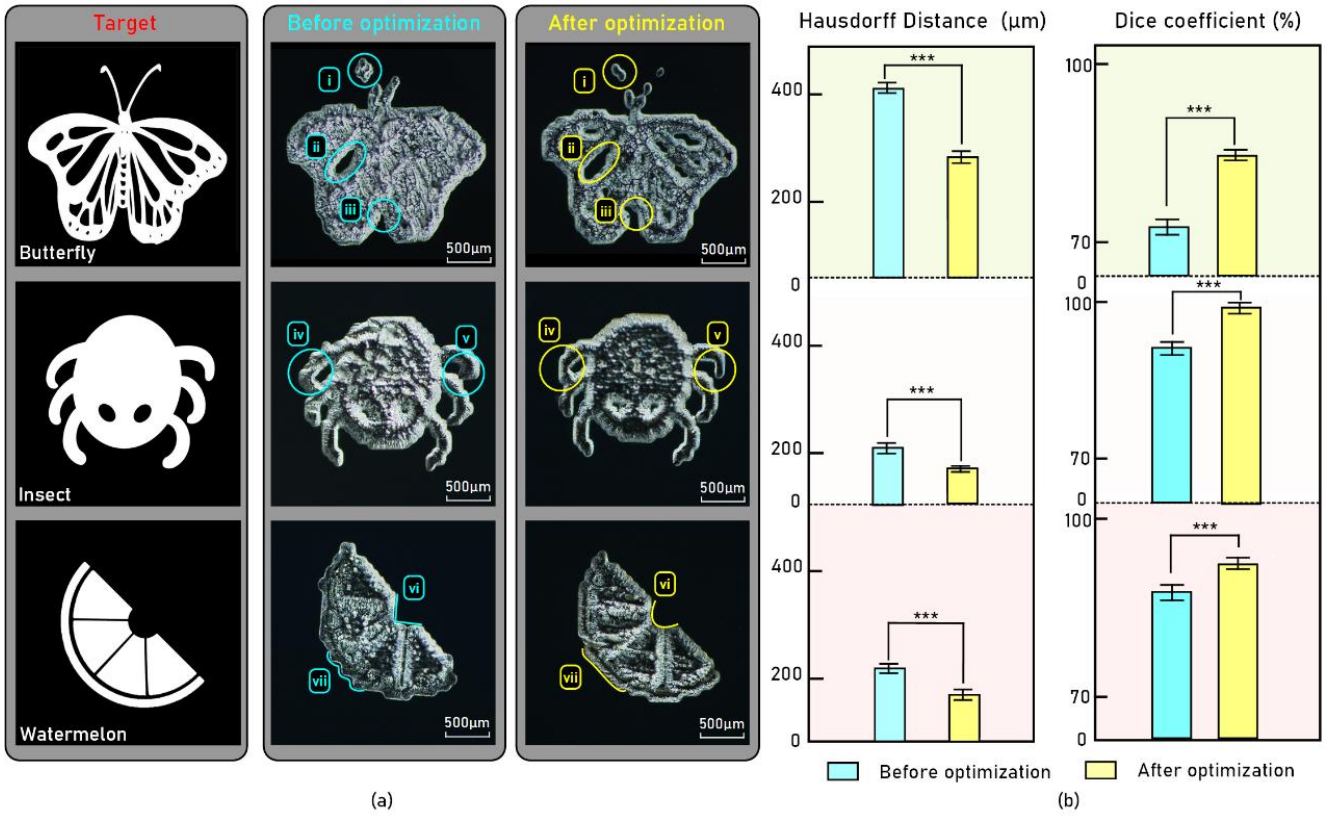


Figure 4-3 Complex and small patterns printed to validate the model's performance: (a) the target pattern and 3D printed structure before and after optimization; (b) Dice coefficient and Hausdorff distance of the printed patterns. Statistical significance was evaluated using a one-way Anova with post hoc Tukey's test (indicates $p < 0.05$ between the two, ** indicates $p < 0.01$, and *** indicates $p < 0.001$).*

4.4 Model applications

To showcase the model's capability in enhancing print quality within the vat polymerization process, two possible applications were presented as examples: the production of a 3D printed filter and the creation of microfluidic channels for demonstration purposes.

4.4.1 3D printed filters

The initial application pertains to producing filters with finer, more precise apertures. Employing vat photopolymerization for creating filters or similar mesh configurations poses a challenge. While generating apertures with diameters covering multiple pixels poses no issues, reducing this diameter to just a few pixels result in traditional techniques struggling to accurately replicate the intended structures, frequently resulting in entirely blocked voids.

In this context, two distinct types of apertures were created: circular and triangular, through the 3D

printing of multi-layer samples (1-3 layers) to fabricate a 2mm*2mm*0.15mm filter. As depicted in Figure 4-4, when the circular hole's size reached approximately 5.2 pixels, the sliced pattern transitioned from a circular to a diamond shape. This transition exacerbated as the hole diameter decreased further, with a 2.6-pixel hole being sliced down to just one pixel (Figure 4-4), ultimately resulting in complete blockage of the intended hole. However, leveraging the trained model enabled the generation of a new set of projection patterns featuring grayscale combinations at the hole's periphery, leading to a more accurate reproduction of the desired shape. This enhancement is evident in the improved accuracy of the final hole, showcasing the system's capability to achieve smaller through-holes that were previously unattainable.

Furthermore, apart from single-layer patterns, the influence of multi-layer structures was also investigated. The holes were evaluated, with partially obstructed holes denoted by the symbol ⊙ and fully blocked ones by ●. It is observable that, with the model's assistance, hole blockages were notably reduced for both single-layer and multi-layer configurations, underscoring the model's effectiveness. A similar enhancement was observed in triangular holes, where the optimized model facilitated the creation of smaller through-holes with enhanced fidelity compared to non-optimized patterns. Even for the 'larger' holes (3-5 pixels), the accuracy of their shapes improved. Nonetheless, it is important to note that while our model succeeded in enhancing the printing success rate for small holes, smaller holes (2-3 pixels) tended to block as the number of layers increased due to the model being trained on 2D data that does not consider inter-layer interactions. In practical printing scenarios, the projected light can penetrate through layers, leading to curing in previously printed layers.

As mentioned in Section 4.3, the varying light intensities at different positions result in inconsistent curing levels of the small holes in the filter. To address this issue, one approach is to opt for an optical system with a more uniform light source. Alternatively, from an algorithmic perspective, calibration of the grayscale projection images at different positions can be achieved by measuring the light intensities at those positions. This calibration can facilitate secondary optimization of the projected images of the small holes at different positions.

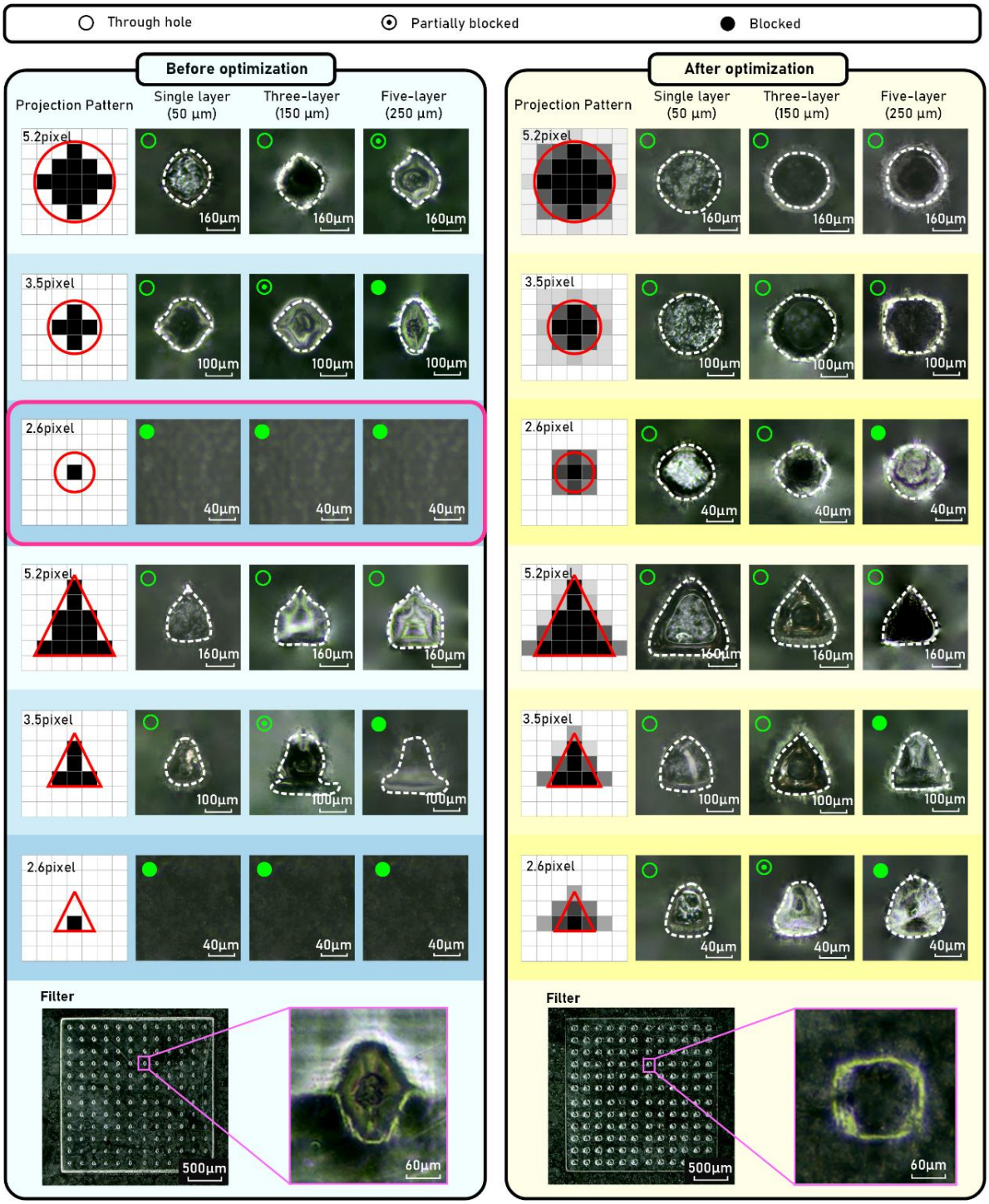


Figure 4-4 The examination of circle (on the left) and triangle (on the right) meshes of varying sizes on a printed filter-like structure. Each row displays the projection pattern alongside its corresponding 3D printed structure, featuring a varying number of layers with each layer's height set at 50 μm. The lower image depicts the filter structure 3D printed using both non-optimized and optimized patterns.

4.4.2 3D printed microfluidic channels

Microfluidic chips [97] represent another application of AM that necessitates high printing accuracy

to achieve small channels effectively. In the context of DLP printing, the formation of small channels (2-4 pixels) can be impeded by the presence of white pixel halos surrounding the channel's sidewalls. To address this challenge, the model was tasked with fabricating a designed miniaturized microfluidic device comprising a reservoir and a network of branched channels with varying channel dimensions, ranging from 7 pixels down to 2 pixels. The 3D printed chip, depicted in Figure 4-5 f-g, measures 9 mm*9 mm*0.25 mm and features embedded channels. Each channel consists of three layers, with each layer having a height of 50 µm, resulting in a total channel height of 0.15mm.

During 3D printing using non-optimized patterns, blockages began to manifest when the channel diameter approached the 5–6-pixel range. As illustrated in Figure 4-5 a-e, when a fluorescent dye was introduced into the microfluidic device, it could only traverse channels with a minimum diameter of 5 or 6 pixels, while channels smaller than this exhibited partial blockages.

On the flip side, when the printing pattern underwent optimization using the trained model, the resulting grayscale pattern demonstrated improved accuracy in reproducing channels. Additionally, the optimized channel width closely approximated the intended design width. As depicted in Figure 4-5h, the unoptimized channel width exhibited a larger deviation from the design, with this deviation magnifying as the width decreased. In contrast, channels produced from the optimized pattern exhibited greater consistency and alignment with the designed width. This improvement can be attributed to the incorporation of greyscale pixels along the channel edges, which helped mitigate the halo effect caused by the previous white pixels and consequently reduced excessive curing responsible for channel blockages.

Enhancing printing precision is crucial for 3D printed microfluidic devices as it enables the fabrication of more compact devices with enhanced flow control. However, achieving this level of precision is not a straightforward task, as it involves either investing in a more costly projector or meticulously fine-tuning the distribution of grey pixels, which can be both expensive and time-consuming. In contrast, leveraging ML techniques to forecast grayscale projection images facilitates the printing of intricate details around 100 µm while streamlining the overall printing process.

When attempting to showcase the cross-section of channels, we encountered fragmentation during the cutting process, indicating potential brittleness of the material used for printing the channels or insufficient thickness of the channels. Moving forward, we will explore options such as changing the

material or increasing the thickness of the channels to further optimize the testing process.

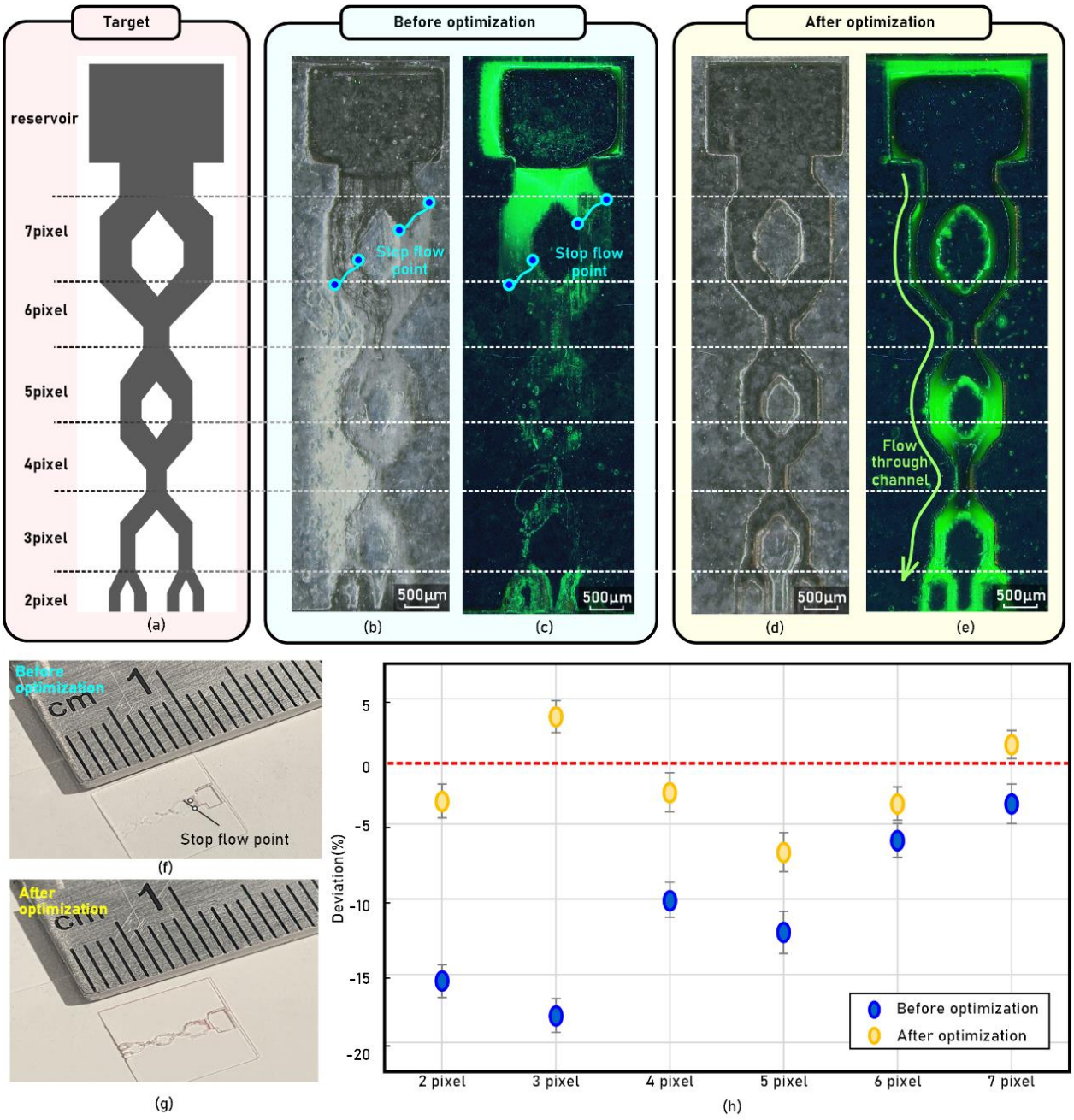


Figure 4-5 Evaluation of the 3D printing process in fabricating microfluidic channels: (a) depicts the channel pattern design; (b) shows the 3D printed channel prior to model optimization, while (c) exhibits the blockage of the channel in the printed structure when fluorescent ink was injected; (d) showcases the 3D printed channel post model optimization, with (e) displaying successful channel creation upon injection of fluorescent ink. The appearance of the device before and after optimization is presented in (f) and (g) respectively, while (h) illustrates the deviation of the 3D printed channel width from the theoretical width.

4.5 Model compatibility

To assess the adaptability of the ML model to various ink types, the scope of the 3D printing process

was expanded to include two additional inks: Resin 'B' (White, Brittle) and Resin 'C' (Grey, ABS-Like) (See Figure 4-6 right). Given the differing reactivities of these inks, adjustments to the intensity of the grayscale pixels in the optimized pattern were required. The standard 3D printing parameter used during training was chosen as the reference, and a crosshair map was designed to determine the necessary intensity adjustments for alternative inks. This crosshair map featured an array of 256 crosshairs with grayscale values spanning from 0 to 255 (see Figure 4-6 left).

For the control ink utilized in model training, the last successfully printed crosshair had an intensity of 96 (see Figure 4-7 left). When applying the same 3D printing parameters to the two additional ink options, it was observed that the crosshair disappeared at intensities of 82 and 69 for Resin 'B' and Resin 'C', respectively (see Figure 4-7 middle and right). Consequently, the overall intensity of the optimized grayscale pattern was adjusted for each resin. As illustrated in Figures 4-8 b-d, following the intensity adjustments of the optimized grayscale patterns, the trained model exhibited a consistent level of optimization across different ink formulations.

During the cleaning of the printed crosshair map, some crosshairs with low grayscale values detached. This was due to their insufficient curing caused by the low grayscale values, resulting in poor adhesion to the release film. Additionally, we observed varying levels of detachment among crosshairs printed with different inks, although there were differences in both location and quantity. This variability was considered a common phenomenon. Shallow curing, which is not conducive to the printing process, does not significantly impact the experiment, making the detachment of the crosshairs not a major concern.

Evaluation of the Dice coefficient for patterns 3D printed using various resins (depicted in Figure 4-8 e) indicated that by adapting the pattern's intensity according to the crosshair map, similar optimization levels were achieved for each resin. This indicates that, post-calibration, the model can effectively accommodate different ink formulations without requiring a full retraining of the model.

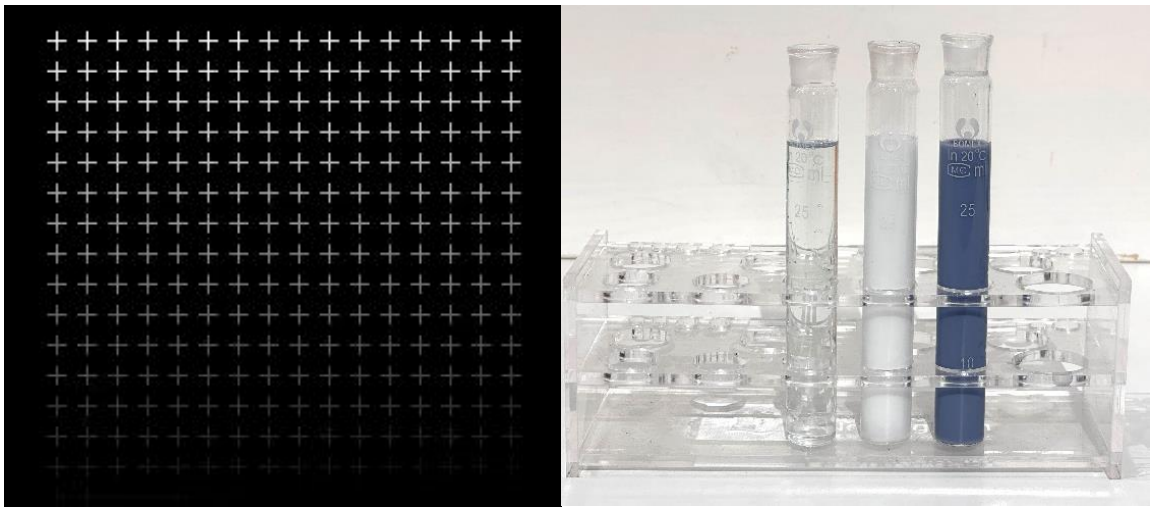


Figure 4-6 A cross-hair map alongside three distinct materials: Resin 'A' on the left, Resin 'B' in the middle, and Resin 'C' on the right.

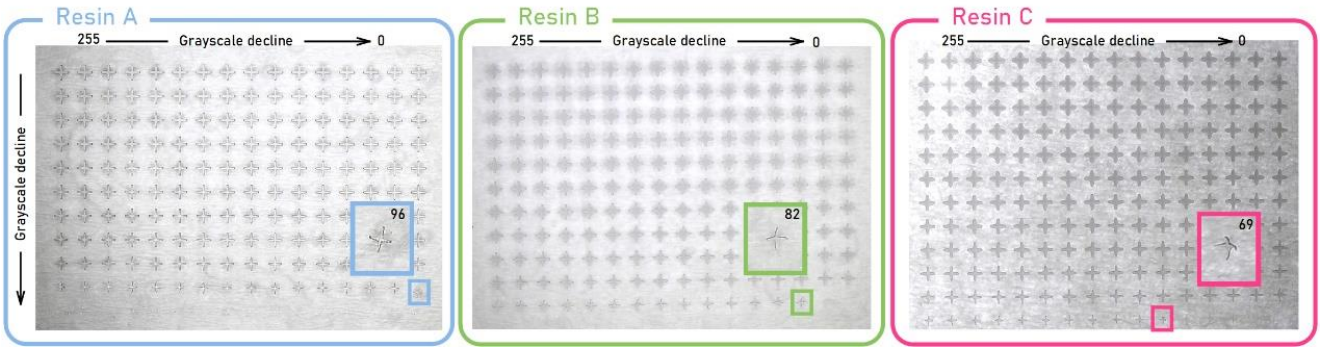


Figure 4-7 A cross-hair map is depicted, showcasing 3D printed models created using three distinct materials: Resin 'A' (Transparent, Brittle), Resin 'B' (White, Brittle), and Resin 'C' (Grey, ABS-Like).

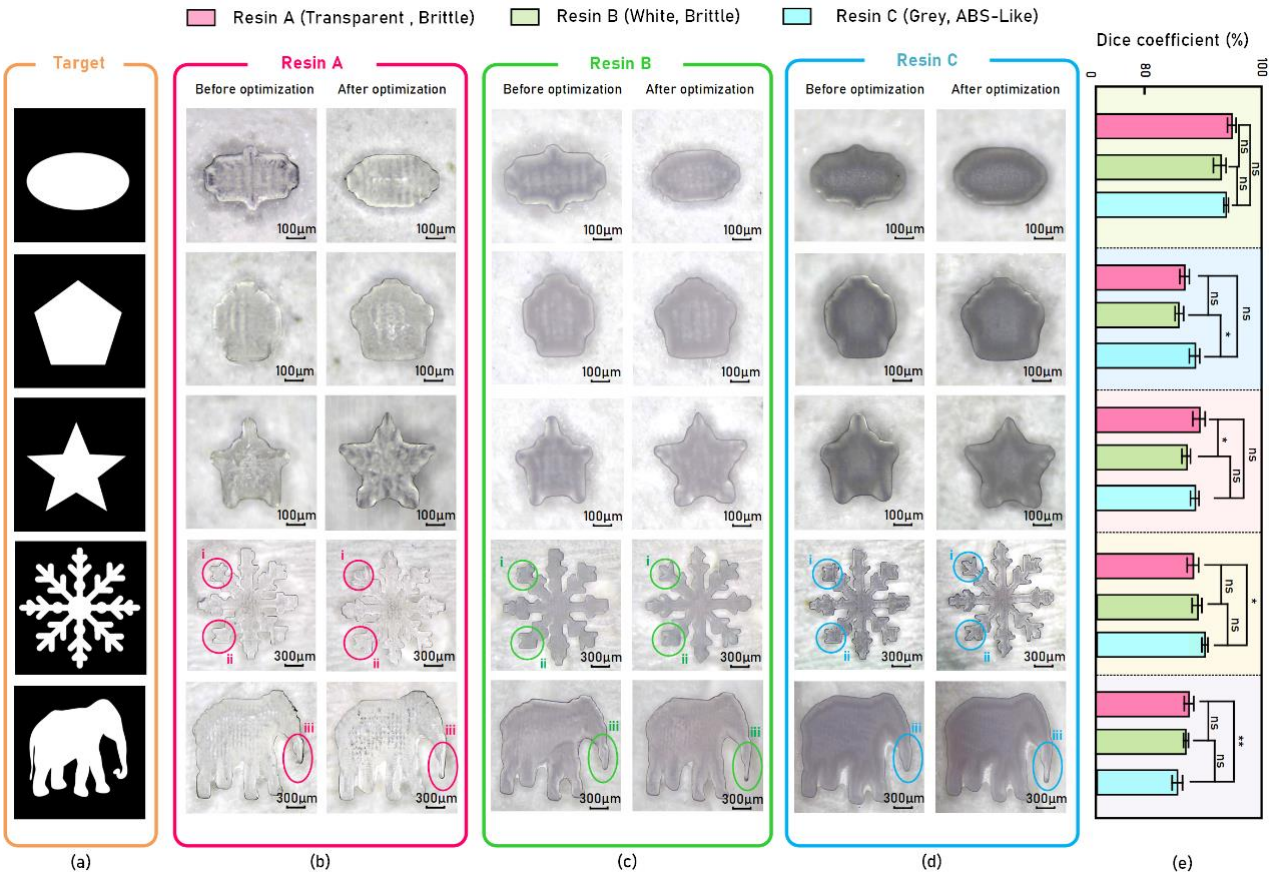


Figure 4-8 The performance evaluation of the machine learning model across various ink formulations: (a) Depiction of the target pattern; (b-d) Comparative analysis of 3D printed structures using different inks, with (b) Resin 'A' (Transparent, rigid), (c) Resin 'B' (White, rigid), and (d) Resin 'C' (Grey, ABS-Like); (e) Calculation of the Dice coefficient for the optimized and intensity-adjusted patterns printed with different ink formulations, along with the results of the significance test. The statistical significance was determined through a one-way Anova followed by a post hoc Tukey's test (indicates $p < 0.05$, ** indicates $p < 0.01$, and *** indicates $p < 0.001$).*

Chapter 5 Conclusion

This thesis introduces a novel approach to data collection and processing for training an ML model based on U-Net, aimed at improving the accuracy of vat photopolymerization in 3D printing. The study encompasses various stages, including the collection of experimental data (Section 3.1), data processing (Section 3.2 and 3.3), data augmentation (Section 3.4), model training (Section 3.5), and assessment of model performance (Section 3.6). Subsequently, the efficacy of the model is validated through its application in real-world scenarios (refer to Section 4).

Through the utilization of a regularized chessboard pattern and an automated calibration technique, we have managed to reduce the training time of the model significantly by a factor of five. The efficacy of this approach is evidenced by the reduction in feature sizes of 3D prints and the enhanced throughput in fabricating intricate structures like filters and microfluidic channels, all without requiring any hardware modifications. The model's ability to adapt to different materials is confirmed through a straightforward calibration process.

When compared to simulation-based methods, we observe that the ML approach is able to achieve around 90% of simulation-based method (92.89±5.2% vs 95.45±1.86%)[1], which could be further improved by introducing more experiment dataset. In addition, our method can reduce the size of printed features as small as 2.6 pixels. The methods in the literature can only print circular features as small as 3.5 pixels. Each method has its own set of advantages and drawbacks. While simulations can capture the fundamental physics of a process and can be easily adjusted to accommodate changes, they typically require an inverse approach to determine optimal. On the other hand, ML methods streamline this analysis process, enabling rapid analysis of large datasets, identification of patterns for optimizing DLP printing, acceleration of iteration speed, and potentially a reduction in the number of experimental trials needed. Although simulation methods offer precise optimization results by simulating material behavior and interactions with light, they demand in-depth analysis of the simulated printing process, which can be time-intensive [1]. Furthermore, ML models can adapt to various materials and environmental conditions through calibration, offering flexibility in optimizing printing accuracy for diverse scenarios. In contrast, physical simulations necessitate substantial computational resources and extensive simulations, limiting their real-time usability in optimizing DLP printing processes.

Nevertheless, this approach does have limitations. Due to the limited training data in 2D, challenges arise when dealing with very small or sharp structures or multi-layered prints. It is speculated that expanding the dataset to encompass a wider range of materials (e.g., hydrogels) and shapes (e.g., training on non-chessboard patterns) could further enhance the effectiveness of our approach.

5.1 Research outcomes

The main research findings of this thesis can be summarized as follows:

(i) Data acquisition and processing strategy: A regularized chessboard pattern was devised to optimize the edges produced by grayscale pixel combinations, alongside an automated calibration approach for streamlined training data collection. This involved incorporating edge detection and error elimination procedures, resulting in a fivefold reduction in model training time and enhanced accuracy.

(ii) Data augmentation technique: CGANs were employed to augment the initial 300 data sets to 1200 data sets. The Dice coefficient similarity between the generated images and the authentic experimental images was measured at 89.7±2.8%, demonstrating a significant reduction in the necessity for extensive data collection. This approach fulfilled the requirement for heightened fidelity and successfully achieved the desired outcome.

(iii) Grayscale optimization model: Leveraging ML models for automated grayscale pixel prediction enhances the process of optimizing printed images, facilitating a reduction in the dimensions of printable features and enabling the creation of circular and triangular apertures as small as 2.6 pixels. The optimization of the model showcased substantial enhancements in throughput for fabricating 3D printing filters and microfluidic channels, all accomplished using identical hardware. Furthermore, experiments conducted with three distinct materials underscored the ease with which findings from one material set could be transferred to another through calibration, thereby validating the model's inherent versatility and applicability across various material types.

5.2 Future works

Based on the findings of this thesis, we present the following future prospects:

(i) Training data optimization: The training data utilized in this study predominantly consists of chessboard patterns and square features. While the model trained on such data exhibits superior optimization performance for simple patterns, its effectiveness diminishes when dealing with complex patterns. Hence, enhancing the diversity of training data is recommended to bolster the model's performance across intricate patterns and scenarios.

(ii) Expanding application scope: Beyond the optimization of two-dimensional images and basic three-dimensional structures, the proposed ML method can be further explored for a wider range of applications involving intricate three-dimensional structures. This includes optimizing features

such as steps on spherical surfaces at the pixel level and enhancing lens surfaces with varying refractive indices.

Chapter 6 Reference

[1] Montgomery, S.M., et al., Pixel-Level Grayscale Manipulation to Improve Accuracy in Digital Light Processing 3D Printing. Advanced Functional Materials, 2023: p. 2213252.

[2] You, S., et al., Mitigating scattering effects in light-based three-dimensional printing using machine learning. Journal of Manufacturing Science and Engineering, 2020. **142**(8): p. 081002.

[3] Gibson, I., et al., Additive manufacturing technologies. Vol. 17. 2021: Springer.

[4] Blakey-Milner, B., et al., Metal additive manufacturing in aerospace: A review. Materials & Design, 2021. **209**: p. 110008.

[5] Moin, D.A., et al., A novel approach for computer-assisted template-guided autotransplantation of teeth with custom 3D designed/printed surgical tooling. An ex vivo proof of concept. Journal of Oral and Maxillofacial Surgery, 2016. **74**(5): p. 895-902.

[6] Moiduddin, K., et al., Structural and mechanical characterization of custom design cranial implant created using additive manufacturing. Electronic Journal of Biotechnology, 2017. **29**: p. 22-31.

[7] Sarvankar, S.G. and S.N. Yewale, Additive manufacturing in automobile industry. Int. J. Res. Aeronaut. Mech. Eng, 2019. **7**(4): p. 1-10.

[8] Delic, M., D.R. Eyers, and J. Mikulic, Additive manufacturing: empirical evidence for supply chain integration and performance from the automotive industry. Supply Chain Management: An International Journal, 2019. **24**(5): p. 604-621.

[9] Davoudinejad, A., Vat photopolymerization methods in additive manufacturing, in Additive Manufacturing. 2021, Elsevier. p. 159-181.

[10] Saccone, M.A., et al., Additive manufacturing of micro-architected metals via hydrogel infusion. Nature, 2022. **612**(7941): p. 685-690.

[11] Guerra Silva, R., M.J. Torres, and J. Zahr Viñuela, A comparison of miniature lattice structures produced by material extrusion and vat photopolymerization additive manufacturing. Polymers, 2021. **13**(13): p. 2163.

[12] van der Linden, P.J., A.M. Popov, and D. Pontoni, Accurate and rapid 3D printing of microfluidic devices using wavelength selection on a DLP printer. Lab on a Chip, 2020. **20**(22):

p. 4128-4140.

[13] Peng, S., et al., Tailoring of photocurable ionogel toward high resilience and low hysteresis 3D printed versatile porous flexible sensor. Chemical Engineering Journal, 2022. **439**: p. 135593.

[14] Jaber, S.T., et al., Evaluation of the fused deposition modeling and the digital light processing techniques in terms of dimensional accuracy of printing dental models used for the fabrication of clear aligners. Clinical and experimental dental research, 2021. **7**(4): p. 591-600.

[15] Zhang, J., et al., Digital light processing based three-dimensional printing for medical applications. International journal of bioprinting, 2020. **6**(1).

[16] Burkhardt, C., et al., Lithography-Based Metal Manufacturing of Jewelry and Watch Cases Made from 316L Stainless Steel and Titanium Alloys.

[17] Ford, S. and T. Minshall, 3D printing in education: a literature review. 2016, DOI.

[18] Brognara, L., et al., Foot orthosis and sensorized house slipper by 3D printing. Materials, 2022. **15**(12): p. 4064.

[19] Longhitano, G.A., et al., The role of 3D printing during COVID-19 pandemic: a review. Progress in additive manufacturing, 2021. **6**: p. 19-37.

[20] Oland, G., O. Garner, and A. de St Maurice, Prospective clinical validation of 3D printed nasopharyngeal swabs for diagnosis of COVID-19. Diagnostic microbiology and infectious disease, 2021. **99**(3): p. 115257.

[21] Fei, L., et al., Wearable solar energy management based on visible solar thermal energy storage for full solar spectrum utilization. Energy Storage Materials, 2021. **42**: p. 636-644.

[22] Zhang, X., et al., Acrylate-based photosensitive resin for stereolithographic three-dimensional printing. Journal of Applied Polymer Science, 2019. **136**(21): p. 47487.

[23] Dai, J., et al., Post-processing of DLP-printed denture base polymer: Impact of a protective coating on the surface characteristics, flexural properties, cytotoxicity, and microbial adhesion. Dental Materials, 2022. **38**(12): p. 2062-2072.

[24] Alshamrani, A.A., R. Raju, and A. Ellakwa, Effect of printing layer thickness and postprinting conditions on the flexural strength and hardness of a 3D-printed resin. BioMed Research International, 2022. **2022**.

[25] Zhao, B., et al., Fast Living 3D Printing via Free Radical Promoted Cationic RAFT Polymerization. Small, 2023: p. 2207637.

[26] Younse, J.M., Mirrors on a chip. IEEE spectrum, 1993. **30**(11): p. 27-31.

[27] Montgomery, S.M., et al., A reaction–diffusion model for grayscale digital light processing 3D printing. Extreme Mechanics Letters, 2022. **53**: p. 101714.

[28] Kowsari, K., et al., Photopolymer formulation to minimize feature size, surface roughness, and stair-stepping in digital light processing-based three-dimensional printing. Additive Manufacturing, 2018. **24**: p. 627-638.

[29] Kumar, H., et al., A kinetic model for predicting imperfections in bioink photopolymerization during visible-light stereolithography printing. Additive Manufacturing, 2022. **55**: p. 102808.

[30] Guven, E., Y. Karpat, and M. Cakmakci, Improving the dimensional accuracy of micro parts 3D printed with projection-based continuous vat photopolymerization using a model-based grayscale optimization method. Additive Manufacturing, 2022. **57**: p. 102954.

[31] Liu, J., et al., Intensity modulation based optical proximity optimization for the maskless lithography. Optics Express, 2020. **28**(1): p. 548-557.

[32] Lee, D., et al., 3D microfabrication of photosensitive resin reinforced with ceramic nanoparticles using LCD microstereolithography. J. Laser Micro/Nanoeng, 2006. **1**(2): p. 142-148.

[33] Park, I.B., Y.M. Ha, and S.H. Lee, Dithering method for improving the surface quality of a microstructure in projection microstereolithography. The International Journal of Advanced Manufacturing Technology, 2011. **52**: p. 545-553.

[34] Mostafa, K., A. Qureshi, and C. Montemagno. Tolerance control using subvoxel gray-scale DLP 3D printing. in ASME International Mechanical Engineering Congress and Exposition. 2017. American Society of Mechanical Engineers.

[35] Zhou, C., Y. Chen, and R.A. Waltz. Optimized mask image projection for solid freeform fabrication. in international design engineering technical conferences and computers and information in engineering conference. 2009.

[36] Kuang, X., et al., Grayscale digital light processing 3D printing for highly functionally graded materials. Science advances, 2019. **5**(5): p. eaav5790.

[37] Forte, C.T., et al., Grayscale Digital Light Processing Gradient Printing for Stress Concentration

Reduction and Material Toughness Enhancement. Journal of Applied Mechanics, 2023. **90**(7): p. 071003.

[38] Zhao, B., et al., Design of grayscale digital light processing 3D printing block by machine learning and evolutionary algorithm. Composites Communications, 2022. **36**: p. 101395.

[39] Wang, J., et al., Stereolithographic (SLA) 3D printing of oral modified-release dosage forms. International journal of pharmaceutics, 2016. **503**(1-2): p. 207-212.

[40] Chen, Q., et al., A study on biosafety of HAP ceramic prepared by SLA-3D printing technology directly. Journal of the Mechanical Behavior of Biomedical Materials, 2019. **98**: p. 327-335.

[41] Wallace, J., et al., Validating continuous digital light processing (cDLP) additive manufacturing accuracy and tissue engineering utility of a dye-initiator package. Biofabrication, 2014. **6**(1): p. 015003.

[42] Mostafaei, A., et al., Binder jet 3D printing—Process parameters, materials, properties, modeling, and challenges. Progress in Materials Science, 2021. **119**: p. 100707.

[43] Singh, R., et al., Powder bed fusion process in additive manufacturing: An overview. Materials Today: Proceedings, 2020. **26**: p. 3058-3070.

[44] Jordan, M.I. and T.M. Mitchell, Machine learning: Trends, perspectives, and prospects. Science, 2015. **349**(6245): p. 255-260.

[45] LeCun, Y., Y. Bengio, and G. Hinton, Deep learning. nature, 2015. **521**(7553): p. 436-444.

[46] Voulodimos, A., et al., Deep learning for computer vision: A brief review. Computational intelligence and neuroscience, 2018. **2018**.

[47] Possatti, L.C., et al. Traffic light recognition using deep learning and prior maps for autonomous cars. in 2019 international joint conference on neural networks (IJCNN). 2019. IEEE.

[48] Zhou, X., J. Zhuo, and P. Krahenbuhl. Bottom-up object detection by grouping extreme and center points. in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019.

[49] Wang, G., et al., Interactive medical image segmentation using deep learning with image-specific fine tuning. IEEE transactions on medical imaging, 2018. **37**(7): p. 1562-1573.

[50] Ronneberger, O., P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. in Medical Image Computing and Computer-Assisted Intervention–MICCAI

2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18. 2015. Springer.

[51] Marzola, F., et al., Deep learning segmentation of transverse musculoskeletal ultrasound images for neuromuscular disease assessment. Computers in Biology and Medicine, 2021. **135**: p. 104623.

[52] Anantharaman, R., M. Velazquez, and Y. Lee. Utilizing mask R-CNN for detection and segmentation of oral diseases. in 2018 IEEE international conference on bioinformatics and biomedicine (BIBM). 2018. IEEE.

[53] Chiao, J.-Y., et al., Detection and classification the breast tumors using mask R-CNN on sonograms. Medicine, 2019. **98**(19).

[54] Li, R., et al., DeepUNet: A deep fully convolutional network for pixel-level sea-land segmentation. IEEE journal of selected topics in applied earth observations and remote sensing, 2018. **11**(11): p. 3954-3962.

[55] Bian, X., S.N. Lim, and N. Zhou. Multiscale fully convolutional network with application to industrial inspection. in 2016 IEEE winter conference on applications of computer vision (WACV). 2016. IEEE.

[56] Wang, J. and X. Liu, Medical image recognition and segmentation of pathological slices of gastric cancer based on Deeplab v3+ neural network. Computer Methods and Programs in Biomedicine, 2021. **207**: p. 106210.

[57] Venugopal, N., Automatic semantic segmentation with DeepLab dilated learning network for change detection in remote sensing images. Neural Processing Letters, 2020. **51**: p. 2355-2377.

[58] Goodfellow, I., et al., Generative adversarial networks. Communications of the ACM, 2020. **63**(11): p. 139-144.

[59] Mirza, M. and S. Osindero, Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784, 2014.

[60] Antipov, G., M. Baccouche, and J.-L. Dugelay. Face aging with conditional generative adversarial networks. in 2017 IEEE international conference on image processing (ICIP). 2017. IEEE.

[61] Nguyen, V., et al. Shadow detection with conditional generative adversarial networks. in

Proceedings of the IEEE International Conference on Computer Vision. 2017.

[62] Qu, Y., et al. Enhanced pix2pix dehazing network. in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019.

[63] Salehi, P. and A. Chalechale. Pix2pix-based stain-to-stain translation: A solution for robust stain normalization in histopathology images analysis. in 2020 International Conference on Machine Vision and Image Processing (MVIP). 2020. IEEE.

[64] Razavi, A., A. Van den Oord, and O. Vinyals, Generating diverse high-fidelity images with vq-vae-2. Advances in neural information processing systems, 2019. **32**.

[65] Rampasek, L., et al., Dr. vae: Drug response variational autoencoder. arXiv preprint arXiv:1706.08203, 2017.

[66] Gao, W., et al., Accelerated discovery of high-performance 3D printing materials using multi-objective active optimization method. Journal of Materials Science, 2024: p. 1-13.

[67] Albuquerque, R.Q., F. Rothenhäusler, and H. Ruckdäschel, Designing formulations of bio-based, multicomponent epoxy resin systems via machine learning. MRS Bulletin, 2023: p. 1-12.

[68] Dabbagh, S.R., O. Ozcan, and S. Tasoglu, Machine learning-enabled optimization of extrusion-based 3D printing. Methods, 2022. **206**: p. 27-40.

[69] Shirmohammadi, M., S.J. Goushchi, and P.M. Keshtiban, Optimization of 3D printing process parameters to minimize surface roughness with hybrid artificial neural network model and particle swarm algorithm. Progress in Additive Manufacturing, 2021. **6**: p. 199-215.

[70] Killgore, J.P., et al., A Data-Driven Approach to Complex Voxel Predictions in Grayscale Digital Light Processing Additive Manufacturing Using U-Nets and Generative Adversarial Networks. Small, 2023: p. 2301987.

[71] Guan, J., et al., Compensating the cell-induced light scattering effect in light-based bioprinting using deep learning. Biofabrication, 2021. **14**(1): p. 015011.

[72] Delli, U. and S. Chang, Automated process monitoring in 3D printing using supervised machine learning. Procedia Manufacturing, 2018. **26**: p. 865-870.

[73] Khan, M.F., et al., Real-time defect detection in 3D printing using machine learning. Materials Today: Proceedings, 2021. **42**: p. 521-528.

[74] Yang, Z., et al., In-situ monitoring of the melt pool dynamics in ultrasound-assisted metal 3D

printing using machine learning. Virtual and Physical Prototyping, 2023. **18**(1): p. e2251453.

[75] Wang, T., et al., In-situ droplet inspection and closed-loop control system using machine learning for liquid metal jet printing. Journal of manufacturing systems, 2018. **47**: p. 83-92.

[76] Creswell, A., et al., Generative adversarial networks: An overview. IEEE signal processing magazine, 2018. **35**(1): p. 53-65.

[77] Gui, J., et al., A review on generative adversarial networks: Algorithms, theory, and applications. IEEE transactions on knowledge and data engineering, 2021. **35**(4): p. 3313-3332.

[78] Zhao, X., et al., Use of unmanned aerial vehicle imagery and deep learning unet to extract rice lodging. Sensors, 2019. **19**(18): p. 3859.

[79] Weng, Y., et al., Nas-unet: Neural architecture search for medical image segmentation. IEEE access, 2019. **7**: p. 44247-44257.

[80] Huang, H., et al. Unet 3+: A full-scale connected unet for medical image segmentation. in ICASSP 2020-2020 IEEE international conference on acoustics, speech and signal processing (ICASSP). 2020. IEEE.

[81] Haralick, R.M., S.R. Sternberg, and X. Zhuang, Image analysis using mathematical morphology. IEEE transactions on pattern analysis and machine intelligence, 1987(4): p. 532-550.

[82] Sarkar, B., et al., Rotational positional error-corrected linear set-up margin calculation technique for lung stereotactic body radiotherapy in a dual imaging environment of 4-D cone beam CT and ExacTrac stereoscopic imaging. La radiologia medica, 2021. **126**(7): p. 979-988.

[83] Zhang, T., S. Wei, and S. Ji. E2ec: An end-to-end contour-based method for high-quality high-speed instance segmentation. in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022.

[84] Perez, L. and J. Wang, The effectiveness of data augmentation in image classification using deep learning. arXiv preprint arXiv:1712.04621, 2017.

[85] Shorten, C. and T.M. Khoshgoftaar, A survey on image data augmentation for deep learning. Journal of big data, 2019. **6**(1): p. 1-48.

[86] Lee, H., et al., Pix2Pix-based data augmentation method for building an image dataset of black ice. Available at SSRN 4044502, 2022.

[87] Lakmal, H. and M. Dissanayake. Illuminating the Roads: Night-to-Day Image Translation for

Improved Visibility at Night. in International Conference on Asia Pacific Advanced Network. 2023. Springer.

[88] Liu, B., et al. Sketch-to-art: Synthesizing stylized art images from sketches. in Proceedings of the Asian Conference on Computer Vision. 2020.

[89] Bottou, L., Stochastic gradient descent tricks, in Neural Networks: Tricks of the Trade: Second Edition. 2012, Springer. p. 421-436.

[90] Liu, T., T. Qiu, and S. Luan, Hyperbolic-tangent-function-based cyclic correlation: Definition and theory. Signal Processing, 2019. **164**: p. 206-216.

[91] De Boer, P.-T., et al., A tutorial on the cross-entropy method. Annals of operations research, 2005. **134**: p. 19-67.

[92] Bharadhwaj, H., K. Xie, and F. Shkurti. Model-predictive control via cross-entropy and gradient-based optimization. in Learning for Dynamics and Control. 2020. PMLR.

[93] Abadi, M., et al. {TensorFlow}: a system for {Large-Scale} machine learning. in 12th USENIX symposium on operating systems design and implementation (OSDI 16). 2016.

[94] Nair, V. and G.E. Hinton. Rectified linear units improve restricted boltzmann machines. in Proceedings of the 27th international conference on machine learning (ICML-10). 2010.

[95] Rame, A. and M. Cord, Dice: Diversity in deep ensembles via conditional redundancy adversarial estimation. arXiv preprint arXiv:2101.05544, 2021.

[96] Huttenlocher, D.P., G.A. Klanderman, and W.J. Rucklidge, Comparing images using the Hausdorff distance. IEEE Transactions on pattern analysis and machine intelligence, 1993. **15**(9): p. 850-863.

[97] Pattanayak, P., et al., Microfluidic chips: recent advances, critical strategies in design, applications and future perspectives. Microfluidics and nanofluidics, 2021. **25**: p. 1-28.