Additive Manufacturing 110 (2025) 104929

Contents lists available at ScienceDirect

Additive Manufacturing

journal homepage: www.elsevier.com/locate/addma

Convolutional autoencoder frameworks for projection multi-photon 3D printing

Ishat Raihan Jamil [a,b], Jason E. Johnson [a,b], Xianfan Xu [a,b,*]

[a] School of Mechanical Engineering, Purdue University, 585 Purdue Mall, West Lafayette, IN 47907, USA
[b] Birck Nanotechnology Center, Purdue University, 1205 Mitch Daniels Blvd., West Lafayette, IN 47907, USA

ARTICLE INFO

Keywords:
Projection 3D printing
Multi-photon polymerization
Micro-nanoscale 3D printing
Convolutional neural network
Autoencoder
Deep learning

ABSTRACT

Projection multi-photon 3D printing is an emerging technique for fabricating micro-nano structures at exceptionally high speeds. It leverages the use of a digital micromirror (DMD) to project and print entire 2D layers at once, offering higher throughput and scalability than conventional point-by-point laser scanning. While two photon polymerization is widely regarded as an outstanding method for achieving high dimensional accuracy at the nanoscale, the projection aspect introduces a new set of challenges, such as under-printing due to oxygen inhibition. The inherently complex photopolymerization dynamics make it difficult to model and simulate efficiently. To address this, we introduce a data-driven methodology employing deep learning to build a surrogate model of the printing process and an inverse model for 2D DMD pattern optimization to achieve desirable printed shapes. By printing diverse shapes morphed by various parametrization schemes, we built a dataset for training convolutional encoder-decoder (autoencoder) neural networks. The trained surrogate accurately maps input DMD patterns to their final printed geometries, capturing nonlinearities introduced by process physics. Inverting the inputs and outputs further enabled us to train an inverse model for generating pre-compensated DMD patterns to print desirable target geometries. Experimental findings demonstrate that this deep learning approach accurately predicts printed outputs and enhances dimensional accuracy in the printing of 2D layers. Our results reveal a viable approach to overcome inhibition-induced constraints, enabling more accurate projection-based multi-photon printing at the micro and nanoscale.

1. Introduction

Additive manufacturing (AM) has transformed the way complex three-dimensional (3D) structures are fabricated, offering the ability to prototype intricate geometries with exceptional precision, all while reducing material waste and overall production costs [1]. While prevalent macroscale additive manufacturing (AM) techniques such as fused deposition modeling (FDM), powder bed fusion (SLM/SLS), and direct energy deposition (DED) face limitations in feature size and surface quality, alternative methods like stereolithography (SLA) and digital light processing (DLP) provide improved resolution and finer achievable feature sizes. Specifically, digital light processing (DLP) based SLA systems improve feature fidelity by projecting patterned light onto photosensitive materials, triggering localized photopolymerization reactions [1,2]. These light-driven techniques leverage the fundamental interaction between photons and matter to initiate precise, layer-by-layer curing of photopolymers [3].

Multi-photon lithography (MPL) is established as a leading strategy for micro- and nanoscale additive manufacturing, achieving nanometer feature sizes [4]. MPL, including two-photon polymerization (TPP), leverages the simultaneous absorption of two (or more) photons of identical frequency from a high-intensity, ultrafast laser pulse, focused onto a small volume of photosensitive resin. This nonlinear phenomenon, enabled by the high photon densities within sub-femtoliter regions of a focused ultrafast laser [5], can produce feature sizes below the diffraction limit [6]. Such capabilities have propelled TPP into widespread use across multiple disciplines, including biomedical engineering [7], micro-robotics [8], micro-optics [9], and metamaterials [10]. However, the point-by-point scanning of nanometer scale voxels inherently constrains throughput and hinders scalability. Various efforts have sought to boost the fabrication speed and output capacity of TPP, for example, by employing holographic techniques that divide a single laser beam into multiple parallel beams. These methods, however, are generally limited to repetitive structures [11,12]. Projection

---

* Corresponding author at: School of Mechanical Engineering, Purdue University, 585 Purdue Mall, West Lafayette, IN 47907, USA.
E-mail address: xxu@ecn.purdue.edu (X. Xu).

https://doi.org/10.1016/j.addma.2025.104929
Received 28 April 2025; Received in revised form 10 August 2025; Accepted 12 August 2025

micro-stereolithography (PμSL) addresses this limitation by utilizing digital light processing with a spatial light modulator (SLM) [13] or a digital micromirror device (DMD) [14], which shapes and projects an intense laser pulse to trigger photopolymerization across an entire 2D layer with a microscale feature size.

The recently developed spatiotemporal focusing of femtosecond laser pulses ensures that only a sub-micrometer-thick target print plane receives sufficient laser intensity to initiate two-photon polymerization, allowing printing at micro- and sub-micrometer scales [15,16]. Conversely, printing full 2D slices incurs challenges such as dimensional inaccuracies due to oxygen inhibition, necessitating additional pre-processing steps [17]. Prior simulation works on spatiotemporal focusing based projection printing focused on numerically modeling the spatiotemporal focusing of the femtosecond pulses [16]. Finite element modeling (FEM) was also employed to explore projection-based TPP on relatively simple nanowires [18]. Nevertheless, numerous theoretical and empirical models formulated for MPL [19–21] suggest the significant difficulty of simulating 3D photon-initiated polymerization for arbitrary geometries, given the numerous process parameters involved. On the other hand, data-driven techniques, encompassing machine learning (ML) and deep learning (DL), have gained traction for optimizing 3D printing processes. While a few studies have applied DL to MPL, these efforts are often centered on limited cases, such as classifying printability. For instance, Pingali et al. [22] trained a classification neural network (NN) on finite element simulations of simple nanowire geometries to assess their printability. They, however, concluded that the surrogate model could not reliably capture fine feature sizes. In another approach, Lee et al. [23] utilized convolutional neural network (CNN) models to identify optimal light dosage and assess part quality on experimental data obtained using a commercial point-by-point laser scanning TPP printer. Additional ML studies have applied Gaussian process (GP) models to address inter- and intra-defects in printed structures. Yang et al. [24] used GP regression to quantify and devise compensation schemes aimed at enhancing geometric compliance. Recently, the Bayesian optimization framework [17] was employed to enhance the geometric fidelity of projection multiphoton nanoscale lithography and demonstrated efficiency by requiring fewer data points per shape. However, its generalizability to other geometries requires further investigation.

Projection-based multiphoton 3D printing holds great potential for fabricating 3D nanostructures with exceptional precision and speed. However, attaining high-fidelity prints continues to be time-consuming due to the complex, layer-by-layer optimization required for each individual 2D projection pattern slices. In this study, we present a comprehensive deep-learning framework designed to enhance the precision and efficiency of projection-based multiphoton printing. Specifically, we introduce a more elaborate parameterization strategy that is built upon the parameterization scheme introduced in [17], alongside an expanded shape catalog for generating 2D DMD input patterns. We also implement a custom optical data acquisition procedure to capture all relevant process parameters. By embedding patterns and print features directly within the images, we aim to establish a more robust image-to-image deep convolutional neural network framework. This approach leverages an encoder to deconstruct the input pattern into a compact vector representation before subsequently being reconstructed by a decoder into a processed representation of the input data [25]. Such architecture is formally referred to as an autoencoder (AE). Traditionally, variations of this network have been primarily employed in biomedical image segmentation [26,27], anomaly detection [28] and object recognition [29]. Here, we adapt the autoencoder paradigm to a supervised learning context, using 2D DMD patterns as inputs and corresponding printed masks as outputs. We developed custom autoencoders built upon popular image classifier designs that were originally intended to address the issues associated with deep neural networks (DNNs), such as vanishing gradients [30]. UNet-inspired skip connections [26] are also incorporated to improve the image reconstruction resolution and optimize the surrogate model. We further demonstrate that by inverting the inputs and outputs to the model, we can train an inverse model to generate pre-compensated DMD patterns to print desirable target geometries with high-dimensional fidelity. This work establishes a novel deep learning-based framework that directly addresses key bottlenecks in projection multiphoton printing, paving the way for more reliable and scalable additive manufacturing at the nanoscale.

2. Methods

2.1. Experimental setup for projection multi-photon printing

A schematic of the projection multiphoton 3D printing configuration utilized in this study is depicted in Fig. 1(a). A Ti:sapphire regenerative amplifier (Coherent Legend Elite Duo USX) produces an 800 nm central wavelength laser beam with a 30 nm bandwidth and a 50 fs pulse duration at a repetition rate of 5 kHz. The beam is routed via a πShaper (AdlOptica πShaper 12_12_TiS_HP), which transforms the Gaussian spatial profile into a flattop distribution, enabling uniform illumination of the 0.45-inch diagonal micromirror array of the DMD (DLP4500NIR). The incidence angle on the DMD surface is established at 24° from the normal. The dynamically patternable device consists of more than a million micromirrors configured in a diamond arrangement, each capable of tilting at an angle of ±12° to denote on/off states. This enables selectively reflecting laser light into the photoresist and thereby create a patterned beam. The one-dimensional (1D) diffraction inherent to the DMD causes the laser pulse to be spectrally dispersed, resulting in temporal pulse elongation and a reduction in peak intensity. An $f = 180$ mm achromatic doublet (Thorlabs AC508–180-AB) in conjunction with a dichroic mirror (Eskma Optics 045–800) collects these scattered components and routes them to the back aperture of a 1.49 numerical aperture (NA) Nikon 100 × objective lens affixed to a piezo-based unidirectional objective scanner (PI PIFOC P-725.4CD). The resin-immersed objective lens concentrates the beam onto the print plane, causing the dispersed wavelengths to reconverge, thus temporally compressing the pulse and reinstating high photon density. This spatiotemporal focusing improves confinement of the pulse intensity to thin layers inside the photoresist. The resin consists of a (2*E*,6*E*)-2,6-Bis (4-(dibutylamino)benzylidene)-4-methylcyclohexanone (BBK)[31] photoinitiator, at 0.38 mol% in PETA monomer as previously reported [16,17]. The mixture is placed in drops onto a microscope glass slide (Fisher Scientific Cat. No. 125441), which has been pre-cleaned by wiping with acetone, followed by consecutive 5-minute baths in alconox and acetone, and then dried with nitrogen gas.

After the printing process, the samples are immersed in a magnetically stirred SU-8 bath at 1150 rpm for about 20 min, followed by a short immersion in isopropanol and drying under a nitrogen stream. Subsequently, the structures are sputter-coated with an Au/Pd mixture and reintroduced to the same experimental setup for imaging, as indicated in Fig. 1(b). This metallic coating improves the contrast between the glass substrate and the three-dimensional printed microstructures in brightfield transmission microscopy [17]. A 660 nm LED source (Thorlabs M660L3-C3) is focused onto the sample plane using a Nikon 100 × ELWD objective (NA = 0.8). The transmitted light is demagnified using a 0.9 NA Olympus 100 × objective that is affixed to the piezo objective scanner. It is then channeled to a CMOS camera (FLIR GS3-U3–32S4M) for image acquisition using a 50/50 beam splitter (Thorlabs BSW27) and an $f = 100$ mm lens (Thorlabs LA1509-B). The photos of the printed structures are further processed with the OpenCV Python package [32]. Otsu's thresholding method [33] is subsequently used to perform binarization, which calculates the optimal threshold level by maximizing inter-class variance pixel intensities in the bimodal grayscale histogram, thereby segmenting the object in question from the background. Prior research benchmark [17] comparing measurements from this brightfield microscopy apparatus to scanning electron microscopy (SEM) estimations of 2–13 μm patterns printed on the same

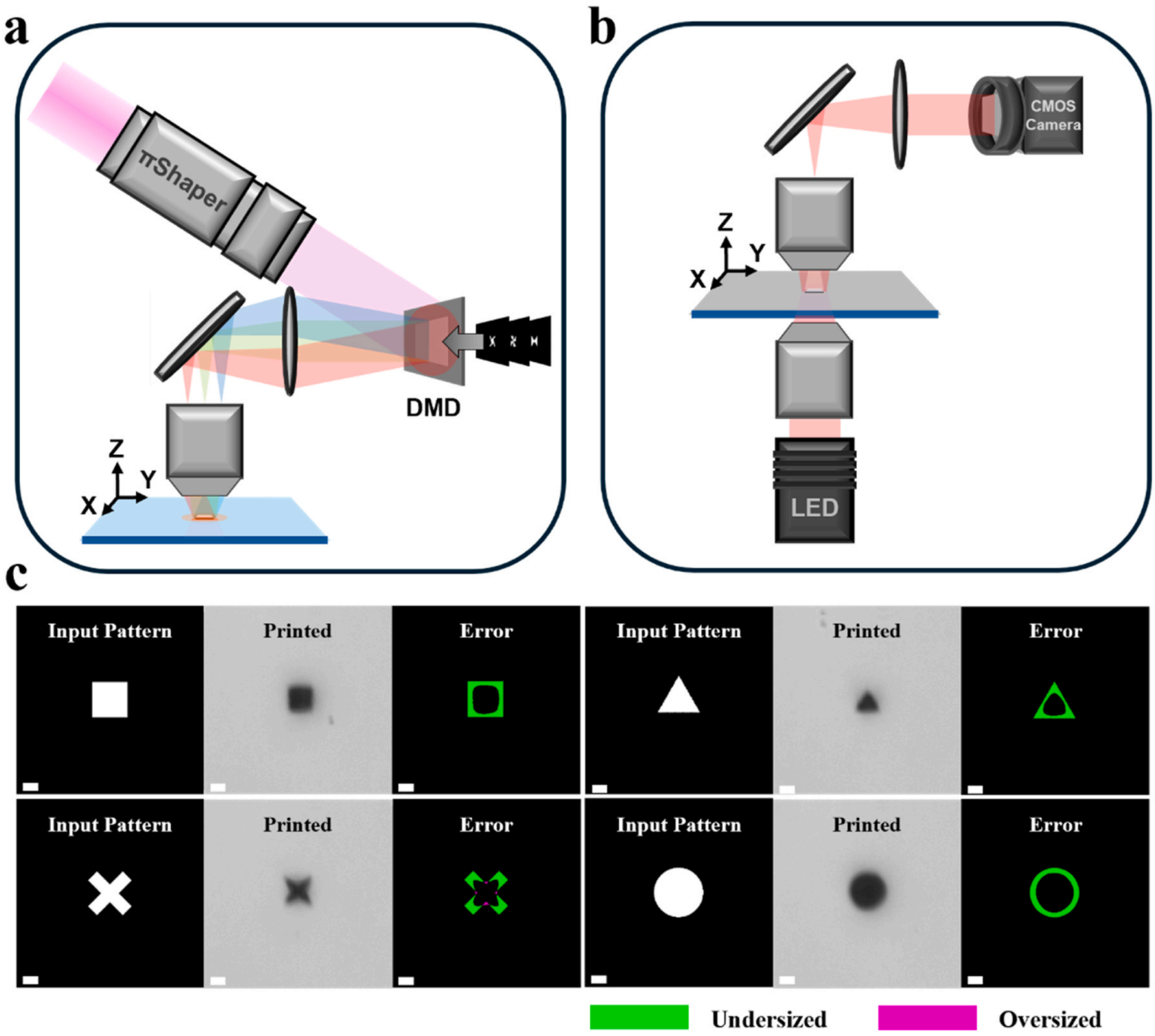

Fig. 1. Overview of the multiphoton projection printing workflow. a. Schematic depiction of the projection printing process, in which a high-intensity femtosecond πshaper transformed flat-top laser beam is shaped by a digital micromirror device (DMD) and projected into a photosensitive resin. The non-linear nature of two-photon absorption enables polymerization to occur only at the region of highest intensity - the print plane. b. Illustration of the same printing setup configured for microscopic imaging of the post development printed structures. c. Comparison between input DMD patterns and actual printed structures reveals a systematic underprinting of approximately 0.7 μm on each side and more pronounced deviations near corners. The scale bars are 2 μm in length.

projection multiphoton printing system revealed that the optical method generally underestimated SEM bounding-box dimensions by only about 200 nm on average, with a slight overestimation for the smallest features as the diffraction limit is approached. Furthermore, since bright-field microscopy is significantly more cost-effective and less time-consuming, it is used in this study to collect images of printed samples.

To print and photograph thousands of individual microstructures, both operations are automated using a three-axis air-bearing translation stage (Aerotech ABL1000 series), which is controlled by LabVIEW and Aerotech's NVIEW stage management software. To achieve parallel autofocusing, a 632.8 nm, 5 mW Helium-Neon laser beam is guided along the same optical path, and its reflection is directed to an independent camera (ZWO ASI183MC). The system utilizes real-time image processing in LabVIEW to position the piezo objective scanner such that the focal plane of the objective is maintained at the substrate surface. Integrating these technologies enables precise focusing with high throughput during printing and subsequent imaging.

2.2. Dataset generation

Micromirrors are activated to on/off states by a two-dimensional binary image of black-and-white pixels that is provided to the DMD. Native oxygen is consumed for radical generation as the shaped beam from DMD is directed onto the print plane [16,34,35]. Depletion of oxygen is a critical prerequisite for the initiation of multi-photon polymerization [19,20]. Nevertheless, the unconstrained diffusion of oxygen from the ambient environment contributes to the polymerization inhibition at the structure's boundaries [36], ultimately resulting in under-printed regions. In Fig. 1(c), the phenomenon is demonstrated by overlapping and comparing the DMD input target patterns with the corresponding binarized printed masks, highlighting the disparities in the error columns. To create an accurate surrogate for this printing process, a DNN model needs to be guided in understanding how micro- and nanoscale feature qualities impact or correct for dimensional imperfections, ultimately serving as the foundation for efficient inverse modeling.

For training the supervised autoencoder, a set of DMD input patterns and their resulting printed shapes are needed. Fig. 2 depicts nine basic geometric forms—ellipses, rectangles, triangles, pentagons, hexagons, heptagons, octagons, crosses, and multi-edged stars (5–8 vertices)—used to train and evaluate the ML model. These 2D designs are then subjected to radial and corner warping techniques to alleviate the under-printing difficulties [17]. Radial warping expands or contracts the

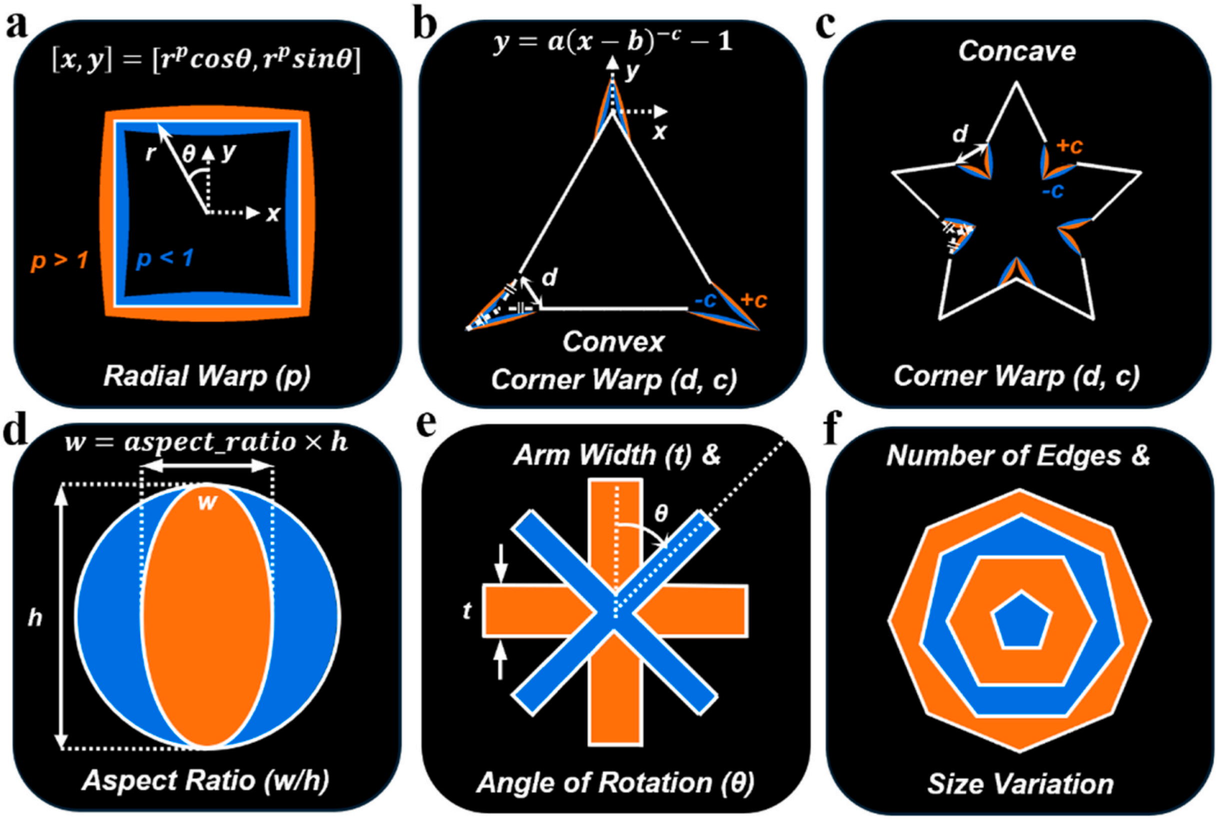

Fig. 2. Parametric variation of fundamental shapes and their geometric parameters. Shown are representative examples of some basic shapes: square, triangle, star, ellipse, cross, and polygons (5–8 edges), along with key parameters such as radial warp, corner warps, aspect ratio, arm width, angle of rotation, and sizes. These parameters were systematically varied using a Sobol sequence, generating a comprehensive dataset for training and validating the deep learning models.

pattern to compensate for the bending of straight edges induced by nonuniform shrinkage and oxygen inhibition. It does this by mapping the radial coordinate system (with the shape centroid as the origin) onto a transformed system, as defined by $[x, y] = [r^p\cos\theta, r^p\sin\theta]$. Here, the radius is elevated to an exponential power p, the radial warping parameter. In polar coordinates, the location $[x, y]$ denotes a point on a shape's perimeter, defined by the radial distance r from the origin and the angle coordinate θ from the origin, as illustrated in Fig. 2(a). Convex corner warping alters the exterior corners in a way that preserves fine details that might otherwise be lost to diffraction and inhibition effects. By designating the corner as the origin, this warping applies the equation $y = a(x-b)^{-c} - 1$. Here, parameters a and b are defined so that the corner warping distance d corresponds to the distance between two pertinent points along the contour. Another point that is the furthest from the origin is selected such that its distance is equivalent to the distance of the contour points from the origin. A curvature profile is then applied between these points, with the parameter c determining the direction and magnitude, as shown in Fig. 2(b). To retain the sharpness of interior edges, the corner warping must be reversed for concave corners for stars and crosses in Fig. 2(c), which exhibit rounding of internal corners due to diffraction and proximity effects [36].

To enhance the generalization of the models, the parametrization scheme was further extended to methodically incorporate a wide range of geometric and dimensional characteristics into the training data. Modifying shape sizes from 3 μm to 10 μm imparts a definitive scale to the model, while aspect ratios between 0.5 and 1.0 rectify distorted dimensions by adjusting the width in relation to height. Angular asymmetry is integrated by rotating shapes from −180° to 180°, whereas thin features are presented by modifying the arm widths of crosses. All the parameters are sampled using a Sobol sequence [37], a quasi-random, low-discrepancy technique that evenly traverses a high-dimensional design space. Table 1 lists the ranges of each of the shape parameters varied using the Sobol sequence. Integrating variations in size, rotation, and skewness directly into the pattern generation scheme for experimentally obtaining data obviates the need for additional data augmentation during post-processing of the printed results. This approach enables deep learning models to properly learn intrinsic spatial, systemic, and printing dynamic effects.

Our goal in this investigation is to determine an optimal shape located within this parameter space. By training the model on simpler shapes scattered over this space, it can gain insight to optimize and tailor individual structural sections, allowing it to generate more complicated DMD input patterns by means of localized design alterations to print desired target geometries. Restricting the investigation to a single photopolymer BBK would further isolate the geometric effects from external variables such as the influence of chemical compositions. For this study, 3390 DMD patterns were generated and printed using the experimental setup at a constant laser power of 24 mW at the print plane with an exposure duration of 10 ms. The exposure condition was previously established for the projection multiphoton printing system,

Table 1
Sobol sequence range of shape parameters.

Parameter	Sobol Range	
	Minimum	Maximum
size ($l_{\mu m}$)	3.0 μm	10.0 μm
radial warping (p)	0.9	1.1
corner warping distance (d)	0 pixels	100 pixels
corner warping curvature (c)	−0.9	0.9
angle of rotation (θ)	−180°	180°
aspect ratio (w/h)	0.5	1.0
star/polygon vertices (n)	5	8
cross arm width (t)	0.5 μm	5.0 μm

where the output qualities remain consistent above a certain threshold dosage (laser power × exposure duration) [17]. At present, this dosage condition provides an ideal printability window that balances effectiveness across a wide range of shape sizes. Below this dosage, small structures do not polymerize reliably. Exceeding this threshold, larger structures are susceptible to over polymerization, resulting in structural warping and possibly detachment from the substrate. The DMD patterns and their associated binarized images of the printed structures constituted a full dataset that was partitioned into an 80:20 ratio for training and validating the deep learning models, respectively. To assess model

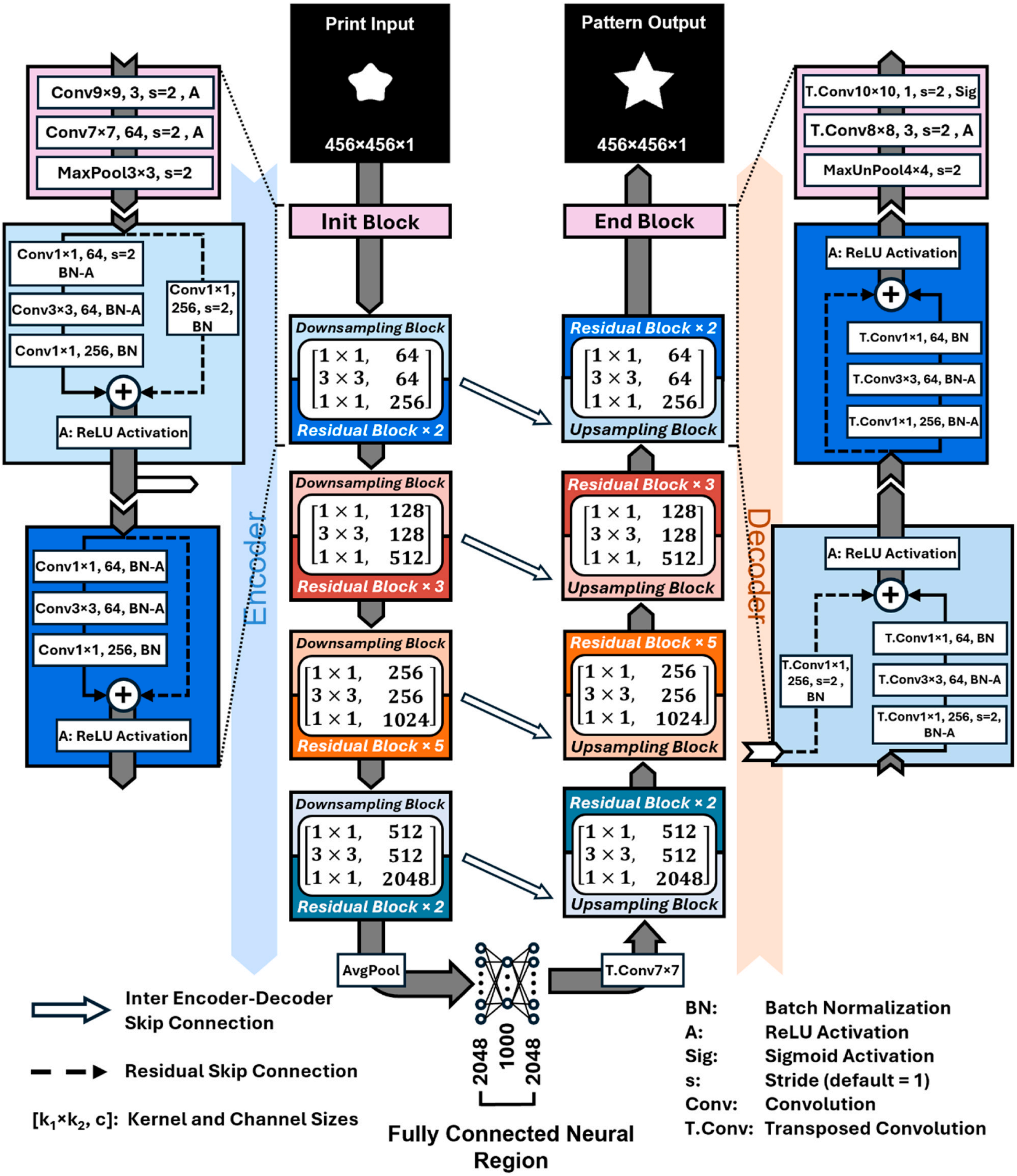

Fig. 3. Architectural overview of the Deep ResNet51-AE: a ResNet-based autoencoder. The model follows an encoder-decoder structure with residual blocks for efficient feature extraction and reconstruction. The Init Block processes the input print pattern using convolutional and max-pooling layers. The encoder consists of downsampling blocks with multiple residual blocks to prevent information loss and improve gradient flow. The bottleneck fully connected region compresses the extracted features before passing them to the decoder, which mirrors the encoder structure with transposed convolutional and upsampling layers. Inter and intra skip connections facilitate information flow across corresponding encoder-decoder layers, aiding in high-fidelity reconstruction. The End Block applies a sigmoid activation to generate the predicted pattern output.

performance, two independent test datasets were generated and printed. Test dataset A was printed at the average print plane power of 24 mW and 10 ms exposure as the training dataset. Test dataset B was printed after improvements to the system had been implemented to reduce the laser pulse duration. Consequently, a lower average print-plane power of 11 mW was required to achieve similar peak intensities at the same exposure period. In total, the two testing sets equate to the size of the validation set and encompass a variety of geometries, including intricate curvilinear designs, fractals, gears, and chiral metamaterial-inspired patterns, as well as basic reference shapes.

2.3. Autoencoder framework

An autoencoder is a neural network architecture trained to learn a compressed representation of input data (the encoding) and subsequently reconstruct that data from it (the decoding). In our case, the autoencoder takes in a 2D DMD input layer pattern and outputs a binarized representation of the printed result. By training on this transformation, the network functions as a surrogate that predicts the printed geometry for a given input pattern, while training on reversed inputs and outputs converts it into an inverse model that generates a compensated DMD pattern for printing any user-defined shape. Fig. 3 illustrates how our ResNet-based autoencoder builds upon residual learning principles to enhance encoder–decoder functionality for image-to-image translation. The model is implemented in Python using the PyTorch library [38]. The encoder resides on the left side, and the decoder on the right side, with both being derived from the ResNet50 framework [30]. Originally developed for image classification, ResNets alleviate vanishing gradients [39], a serious hurdle in training very deep neural networks, by incorporating residual blocks. Additionally, having the DMD pattern and the binary mask of the printed structure on the same camera plane provides flexibility: the network can be inverted by simply interchanging the input and output during training. Consequently, our ResNet51-AE serves as a surrogate model for predicting printed geometry for a given input pattern and as an inverse model to generate pre-compensated DMD patterns for printing any user-defined shape, hence enabling enhanced dimensional precision in multi-photon additive manufacturing.

The encoder and the decoder in Fig. 3 consist of an initialization block (Init Block) and an End block, respectively, followed by four Downsampling/Upsampling – recurring Residual block pairs, with mirroring sets of convolution/transposed convolution operations. The number of times each Residual blocks are repeated is noted within each block pair (for example: *Residual Block* $\times$ *2*). In a forward pass through the model, the encoder convolutions compress the input binary image into a 1D vector representation that is passed onto a fully connected neural network region at the bottom of Fig. 3. The decoder path begins by taking in this latent vector representation and reconstructing a tensor for the transposed convolutions to upsample. The fully decoded (predicted) image at the end of this pathway is then compared to the ground truth binary image (reference). The difference between the predicted and reference images is then used to update the tunable kernel weights (discussed below) during the backpropagation pass across the model. Throughout the training process, several forward and backpropagation passes are made using the images in the training datasets until minimal disparity is achieved between the predicted and reference images.

The exploded views of the first few blocks of the encoder on the leftmost side and the last few blocks of the decoder on the rightmost side are also illustrated in Fig. 3 to provide more details. The Init block begins with a convolution layer $Conv9 \times 9, 3, s = 2, A$ (*kernel size: 9* $\times$*9, channel depth: 3, stride: 2, A: Relu activation*) to convert the input binary image (height: *456 pixels* $\times$ width: *456 pixels* $\times$ channel depth: *1*) at the camera coordinate space to the input size that a standard ResNet-based architecture expects. A similar transposed convolution: $T.Conv10 \times 10,\ 1,\ s = 2$ with a sigmoid activation (sig in Fig. 3) in the End Block at the end of the decoder upsample the final tensor to a ground truth pixel-level image. This avoids the need for rescaling, thereby preserving the spatial fidelity of microstructures and preventing the generation of unwanted artifacts. During a convolution operation, a kernel with dimensions (height: $k_1\times$ *width:* k_2) slides across the input image tensor to extract its hierarchical features. Each kernel contains $k_1 \times k_2$ learnable weights that are updated during the backpropagation step of the training process. The parameter stride (s) determines how many pixels the kernel shifts along each direction of the input tensor. Transposed convolutions on the decoder side in Fig. 3 essentially reverse this operation to upsample its inputs. ReLU (Rectified Linear Unit) activations [40] associated with each convolution/transposed convolution generates non-linearity by zeroing out negative values in the convoluted feature map to aid in learning intricate dependencies (to be explained in Supplemental Note 1). Having the ground truth as a binary image allows the final transposed convolution in the End block in Fig. 3 to use sigmoid activation for calculating a probability for each pixel to be either 1 (white) or 0 (black). The Init and End blocks in Fig. 3 also contain another class of operations called Max-Pooling and Max-Unpooling. Max pooling operates similarly to convolution but only passes on the largest value within each kernel window, essentially downsampling feature maps and reducing computational complexity by incurring no additional learnable weights. Max unpooling uses the spatial location of these maximums to return the upsampled values to the appropriate coordinates while filling all other positions with zeros, thus reconstructing high-resolution outputs from low-resolution inputs.

As the name suggests, the purpose of the Downsampling and Upsampling blocks is to decrease and increase the 2D spatial dimensions of an image tensor using a series of convolutions and transposed convolutions, respectively. The procedures described within each of the four Downsampling/Upsampling - Residual block pairs in Fig. 3 detail the convolutions/transposed convolutions that are executed sequentially. For instance, in the second convolution of the exploded Downsampling block (blue box) on the left side of Fig. 3 $\left(\begin{bmatrix} 1\times 1, & 64 \\ 3\times 3, & 64 \\ 1\times 1, & 256 \end{bmatrix}\right)$: 3×3 describes the size of kernels ($k_1 \times k_2$) that project the spatial information into 64 feature maps/channels (c). The Downsampling blocks are accompanied by a bypass link (dashed arrow) that applies an additional convolution (*Conv1* $\times$*1, c*) to align the shape of the side routed data before elementwise adding it to the output from the other convolutions before applying the activation function. The encoder passes the resulting tensor onwards to the next residual block as well as a copy to the decoder upsampling block on the mirroring side via inter encoder-decoder skip connections (white arrows in Fig. 3). The Residual Blocks as illustrated in Fig. 3 apply similar successive convolutional operations but without changing the spatial tensor shapes. They are also complemented with direct residual skip connections that aid in resolving vanishing gradient problems (explained in Supplemental Note 1) by providing a straightforward pathway for gradients to flow back during the backpropagation steps. On the other hand, the Upsampling block employs a transposed convolution (*T.Conv1* $\times$*1, c*) on the encoder-decoder skip connection data (white arrows) to convey fine details from the encoder to the decoder workflow, similar to an UNet implementation [41]. The autoencoder training process is further enhanced with additional optimization techniques such as batch normalizations (BN) [42] (to be explained in Supplemental Note 1) implemented within each layer.

The subsequent Downsampling/Upsampling – Residual block pairs (red, orange, and teal blocks) in Fig. 3 repeat the processes described above, having the same kernel size aspects (to downsize the 2D spatial resolution in the encoder and upsize in the decoder) but with progressively larger feature maps. The final encoded tensor values are spatially averaged using an average pooling layer, which evaluates the mean value within each local area of the feature map, therefore reducing the spatial dimensions while preserving the most relevant information. The resulting feature maps are then flattened along the channel dimensions to be passed onto a set of dense neural layers containing 2048, 1000, and another 2048 neurons consecutively. This encoder-decoder linking fully connected neural region acts as a bottleneck, forcing the model to only learn the essential features from the input data. A succeeding transposed convolution unflattens the neurons and prepares the tensor for the decoder operations to process. In total, there are 51 operations layers in each half of the encoder-neural-decoder pathways, giving our custom autoencoder the name ResNet51-AE.

3. Results and discussions

3.1. Comparing models

To find an effective autoencoder for both surrogate and inverse models, we analyzed various existing and custom-built autoencoder models. Four architectures: R2 U-Net, AlexNet AE, Deep ResNet35-AE, and Deep ResNet51-AE are benchmarked for surrogate modeling. R2 U-Net [27] is built upon the classic UNet architecture [26] using recurrent residual convolution units that outperformed the UNet in several biomedical image segmentation tasks. AlexNet autoencoder, which was introduced in the supplemental notes in one of our previous works [17] to compare with a Bayesian optimization scheme, was constructed by combining two AlexNets [43] end-on-end with UNet-like skip connections between the encoder and decoder. Deep ResNet35-AE, derived from the ResNet34 network [30], has a similar architecture to Deep ResNet51-AE but with different types and numbers of residual blocks. The Deep ResNet51-AE uses bottleneck residual blocks consisting of three sequential convolutional operations (*Conv1 ×1, c/4 - Conv3 ×3, c/4 - Conv1 ×1, c*) as illustrated in the exploded view of the residual block in Fig. 3 and described in Section 2.3. On the other hand, Deep ResNet35-AE uses the basic residual block type consisting of two consecutive convolutions (*Conv3 ×3, c - Conv3 ×3, c*) in addition to a residual connection. Custom models give us the flexibility of tailoring the layers to meet our specific needs. For all tested architectures, we add additional convolution and transposed convolution layers at the beginning and end to adjust the original image dimensions to the input/output sizes expected by each model, as was described from the Deep ResNet51-AE model in Section 2.3. Additional information on the model architectures can be found in Supplemental Note 2.

Each of the models described above was trained on an Nvidia GeForce RTX 3090 with 24 GB of VRAM using the training dataset, and hyperparameters were tuned using the validation dataset. To compare the performance of these autoencoders, they were inferred without further fine-tuning or re-training on the total test dataset, encompassing test dataset A (generated under the same print conditions as the training dataset) and test dataset B (printed using the improved laser system). Fig. 4 presents the statistics of the analysis. Each sub-figure utilizes the top vertical axis to illustrate the accuracy score (red bars with a maximum at 1.0) and F1 score (orange bars with a maximum at 1.0). These criteria, which are extensively used in classifier evaluations [44] have been repurposed to assess pixel-level predictions in our binary images. Accuracy score indicates the proportion of correctly classified pixels relative to the total pixel count:

$$Accuracy\ Score = \frac{TP + TN}{TP + FP + TN + FN} \tag{1}$$

where TP (true positives) and TN (true negatives) correspond to correctly predicted white and black pixels, respectively, while FP (false positives) and FN (false negatives) refer to incorrect predictions of black-to-white and white-to-black pixels. Accuracy score thus provides insight into the extent to which the predicted morphologies encompass the target area (areal fidelity). By contrast, the F1 score is the harmonic mean of Precision and Recall:

$$F1Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{2}$$

Precision quantifies the accuracy with which a model identifies white pixels, while recall reflects the capability of the model to identify white pixels:

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

In our application, any overestimation of shape edges creates excess white pixels (*False Positives*), whereas underestimation yields missing

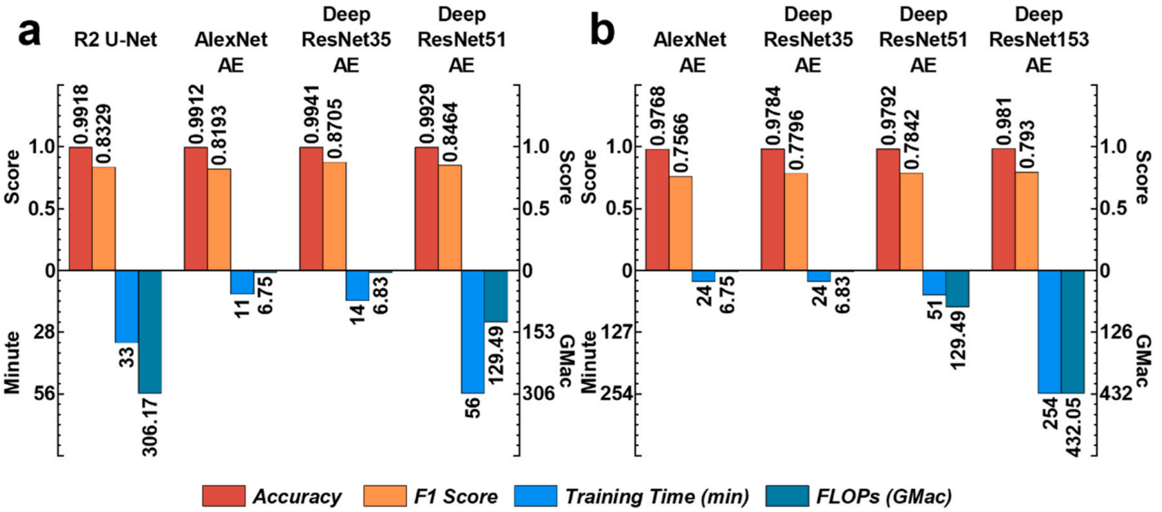

Fig. 4. Performance evaluation of various autoencoder architectures on the test datasets A and B. The top scales represent the accuracy score (measuring areal fidelity) and the F1 score (measuring perimetric precision). The bottom scales represent the training time in minutes and floating-point arithmetic operations (FLOPs) in Giga Multiply-Accumulate Operations (GMacs), with the axis being normalized by their respective maximum values. Longer training times and higher FLOPs imply greater computational and memory costs. a. Surrogate model evaluations compare an established biomedical image-segmentation autoencoder, R2 U-Net, with our custom AlexNet autoencoder (AE), Deep ResNet35-AE, and Deep ResNet51-AE developed by concatenating end-to-end two popular image classifiers of the same name, forming the encoder and decoder segments. **b.** Inverse model evaluations reveal increasing model complexity from AlexNet autoencoder to Deep ResNet153-AE comes at the cost of longer training durations, larger computational and memory requirements, with marginal gain in accuracy and F1 scores. The Deep ResNet51-AE strikes a favorable balance between computational efficiency and predictive performance, making it a practical choice for both surrogate and inverse modeling.

white pixels (*False Negatives*). Consequently, the F1 score estimates perimetric fidelity, i.e., how accurately the predicted edges match the ground truth.

The lower vertical axes in Fig. 4 convey the training period in minutes (blue bars) and the floating-point operations (FLOPs) (calculated using the ptflops library [45]) in giga multiply-accumulate (GMac) operations (teal bars), both normalized by their maximum observed values to allow for a fair comparison of different models. Training time is the product of the length of time required to pass the entire dataset through the model once in each epoch (given the hyper-tuned learning rate and batch size), and the total number of epochs necessary to achieve minimal validation loss. FLOPs, together with model training intervals, give insight into the computational expense necessary to train a deep-learning model. Higher FLOPs indicate higher computational and memory costs.

It is apparent from Fig. 4(a) that the accuracy and F1 score of all 4 surrogate models are comparable, indicating each of them has sufficient model parameters to comprehend the oxygen inhibition dynamics. In terms of computational efficiency, R2 U-Net and Deep ResNet51-AE train much slower than the others, partly due to the higher FLOPs and partly due to higher/lower model capacity to learn from the training data effectively. Despite the cost, Deep ResNet51-AE manages to achieve superior surrogate predictive performance, closely matching that of Deep ResNet35-AE, which consistently exhibits the most optimal surrogate performance among the models evaluated. However, the trivial disparity in Accuracy and F1 scores among the models dictates that any of the models can serve as a surrogate for the printing process.

Fig. 4(b) presents a comparable analysis for inverse modelling and extends the ResNet-based networks to a deeper variant: Deep ResNet153-AE, which is constructed from the ResNet152 network [30]. Here, the deeper models achieve incremental gains in accuracy and F1 score but incur a pronounced increase in computing need (FLOPs and training time). While the most extensive network (Deep ResNet153-AE) exhibits a marginally higher precision, the additional cost may not be justified when resources or rapid iteration cycles are a priority. Conversely, shallower networks (AlexNet AE and Deep ResNet35 AE) reflect the limitations of a lower model capacity in inverse predictions, as evident by their significantly reduced Accuracy and F1 scores.

In general, Fig. 4 exhibits how increasing model complexity improves predictive reliability but heightens the computation cost. An in-depth comparison of all the model prediction/printing results is presented in Supplemental Note 3. Among the evaluated models, Deep ResNet51-AE appears to find a practical balance between Accuracy, F1 score, and computational workload for both surrogate and inverse predictions, indicating that a moderately deep residual architecture can produce robust results without requiring excessive training time or computing power. The prediction latency of the ResNet51-AE model, demonstrated in Supplemental Note 4, implies that once the model weights are loaded into memory, it can potentially optimize each pattern in as little as 0.06 s, facilitating high-throughput printing. Hence, the Deep ResNet51-AE is chosen to be the autoencoder used for this work.

3.2. *Surrogate and inverse predictions*

Fig. 5 depicts the performance of the Deep ResNet51-AE when used as a surrogate model. The DMD input patterns in Columns 1 and 5 (derived from test dataset A) were not used for training or validation of the model, thereby allowing for an unbiased assessment of the model's generalizability. A 2 μm scale bar is included in each image. The resulting polymerized structures for these DMD patterns are shown in the printed columns (Columns 2 and 6) of Fig. 5. It is evident from the figure that the unaltered DMD patterns result in under-printing of approximately 0.7 μm on each side and more at the corners due to an oxygen dead zone as was seen previously in [17]. The prediction columns (Columns 3 and 7) illustrate the autoencoder's optimal reconstruction of the printed shape based on the original DMD input. The error columns (Columns 4 and 8) indicate disparities between the estimated and actual prints. Green contours indicate locations where the prediction is undersized in comparison to the print, whereas magenta contours indicate areas where the prediction is oversized relative to the print. This color-coding facilitates the identification of contours around the shape that diverge from the anticipated boundaries. The small perimeter errors observable in Fig. 5 indicate that the surrogate Deep ResNet51-AE model can effectively predict the printed shape of multiple test patterns with various dimensions.

Perimeter error, the area of the under and over-predicted regions (green and magenta pixels in Columns 4 and 8 in Fig. 5) divided by the

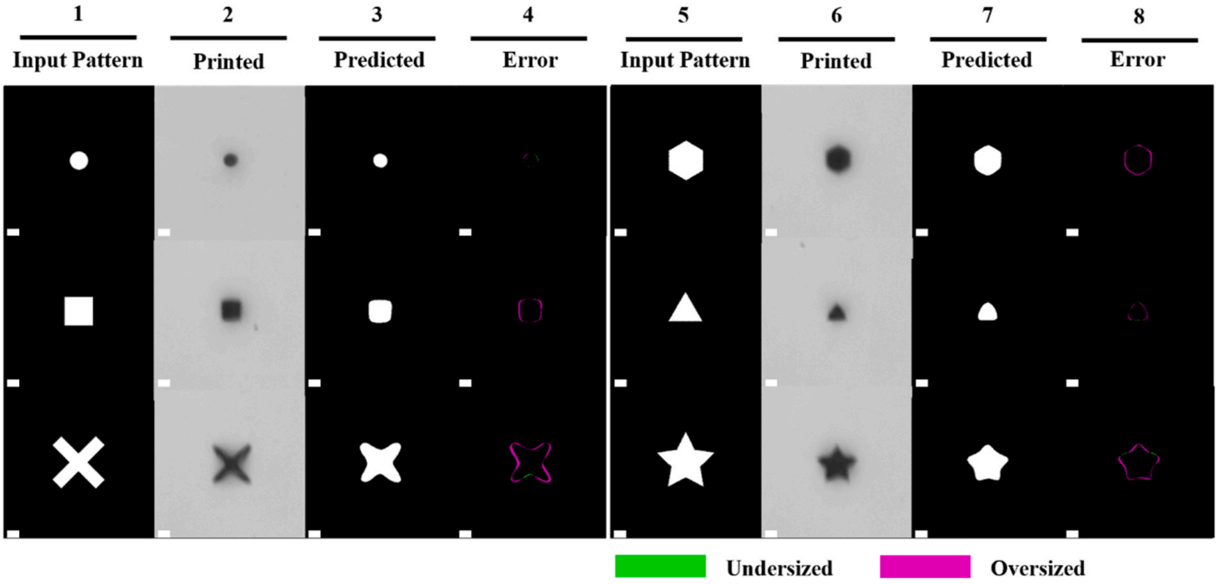

Fig. 5. Qualitative results from the Deep ResNet51-AE surrogate model. Columns 1 & 5 showcase some target patterns derived from test dataset A, which are similar to training data but previously unseen by the surrogate model. Visualization of the corresponding printed results is depicted in Columns 2 & 6. Columns 3 & 7 display the binary printed shapes predicted by the autoencoder. Columns 4 & 8 compare the binarized print masks with the model predictions. The scale bars are 2 μm in length.

perimeter of the printed shape (in Column 2) measured in μm, is used to quantitatively evaluate the accuracy of the surrogate model. For the shapes displayed in Fig. 5, the autoencoder is able to predict print shapes from DMD patterns with a perimeter error of 0.031–0.192 μm, as listed in Table 2. This is comparable to about 1–3 demagnified camera pixels (each pixel size: 62 nm) and falls within the previously determined measurement accuracy [17].

The trained inverse Deep ResNet51-AE model's workflow is illustrated in Fig. 6, which seeks to create pre-compensated DMD patterns (Column 4) for particular target shapes (Column 1) taken from the unseen test dataset A. The under-printing due to oxygen inhibition is again observed for the uncompensated DMD patterns. This is demonstrated by the initial print column (Column 2) and its initial error column (Column 3) when compared to the target patterns from Column 1. The inverse model's optimized patterns (Column 4) are designed to compensate for these dimensional inaccuracies. The optimized prints (Column 5) for these predicted patterns, when compared with the target shapes (Column 1), show a significant reduction in errors (Column 6). It is noteworthy that some of the imperfections visible in Column 6 can be attributed to imaging artifacts such as shadowing or uneven illumination, which can affect visual comparison but not the underlying integrity of the prints. Nonetheless, the substantial reduction in the green and magenta contours illustrates the enhancement that has been achieved with the inverse autoencoder's predicted patterns.

The asymmetrical DMD patterns apparent in Column 4 in Fig. 6 portray a learnt adaptation by the model, rather than a defect. Despite the symmetrical nature of the target shapes, the printing system's inherent numerous asymmetries, such as 1-dimensional diffraction from the DMD micromirrors, laser intensity non-uniformity, and minor optical misalignments, culminate in disproportionate under-printed errors in the printed structures in Column 2. By spatially altering the patterns in a manner that may seem skewed, the model learns to discern and compensate for these system-level biases, thus ensuring that the final printed shapes in Column 5 are closer to the symmetric target.

For the inverse model, the perimeter error is computed by dividing the area of the under and over-printed regions (green and magenta pixels in Columns 3 and 6) by the perimeter of the target pattern (in Column 1) for the initial and optimized structures in Fig. 6. For shape sizes greater than 4 μm shown in Fig. 6, the inverse Deep ResNet51-AE is able to reduce the average perimeter error from 0.502 μm to 0.118 μm, as tabulated in Table 3, which is again within to 1–3 demagnified camera pixels and within the measurement accuracy [17]. For shapes smaller than 4 μm (the circular dot in Fig. 6), the resulting perimeter error improves from 0.362 μm to 0.231 μm at the current fixed power and exposure. It may be possible to enhance the printing of smaller structures by training the autoencoder on variable exposure patterns generated from grey-scaling methods. Nonetheless, Fig. 6 demonstrates that the inverse model, in general, can anticipate a DMD pattern that compensates for oxygen inhibition and other systematic distortions, resulting in a final print that more closely resembles the target geometry.

Fig. 5 and 6, along with Table 1 and 2, collectively demonstrate the dual functionality of the Deep ResNet51-AE architecture. In surrogate mode, it can reliably forecast the final print from an input pattern, while in inverse mode, it can alter the DMD input pattern to significantly reduce the observed dimensional imperfections. The intuitive representation of both undersized and oversized visual outcomes corroborates the previously provided quantitative accuracy and F1 measurements, demonstrating that, despite slight boundary discrepancies, the model continually performs well across diverse shape complexity and sizes. Supplemental Note 5 further verifies the multi-scale predictive capabilities of the Deep ResNet51-AE model architecture by optimizing the same patterns at different length scales.

Table 2
Surrogate model prediction perimeter error on patterns taken from test dataset A.

Shape	Target Size (μm)	Perimeter Error (μm)
ellipse	3.0	0.031
rectangle	6.5	0.120
cross	10.0	0.192
hexagon	6.5	0.090
triangle	6.5	0.087
star	10.0	0.121

3.3. *Complex shape predictions*

The performance of the Deep ResNet51-AE's inverse and surrogate model for more complex target patterns (derived from both test datasets A and B), which are constructed by concatenating multiple simpler shapes, is illustrated in Fig. 7. The complexity of these shapes allows us to assess the models' ability to generalize to novel, previously unexplored (not trained upon) geometries. In addition to the actual printed results and associated errors, each column in Fig. 7 offers a step-by-step visualization of both inverse and surrogate predictions. The feature sizes pertinent to micro- and nanoscale manufacturing are shown by the 2 μm scale bars.

Column 1 in Fig. 7 displays the input target patterns, which are not included in the training and validation sets, therefore providing a thorough evaluation of the model's prediction ability. Columns 2 and 3 show the printed results of the target pattern and the corresponding initial errors, respectively. The green overlays indicate underprinted regions, and magenta overlays denote overprinted areas, as previously described. These columns demonstrate the degree of dimensional distortion often resulting from oxygen inhibition or other process-related factors. Column 4 presents the optimized DMD pattern produced by the inverse model, aimed at mitigating the errors in Column 3. Columns 5 and 6 show the printed results of these pre-compensated patterns, along with the binarized error masks when compared to the target shapes from Column 1. The qualitative analysis indicates that the new prints much more closely align with and approximate the intended (target) shape. From a quantitative standpoint, the average perimeter error in Fig. 7 is reduced from 0.483 μm in Column 3 to 0.158 μm in Column 6, as listed in Table 4. Again, some of these errors can be associated with imaging imperfections such as shadowing or unbalanced lighting, which may impair visual comparison but not the prints' overall integrity.

Column 7 in Fig. 7 presents the surrogate model's estimations based on the optimized patterns of the inverse model shown in Column 4. It illustrates the surrogate's estimations regarding the anticipated appearance of the final structures with the optimized patterns. Column 8 highlights the minimal discrepancies between the surrogate-inverse predictions and the target patterns, with a mean perimeter error of about 0.151 μm as inferred from Table 4. This facilitates a direct assessment of the model's accuracy in predicting perimeter dimensions for intricate, multi-feature designs.

In general, Fig. 7 illustrates that the Deep ResNet51-AE performs well with complex shapes, enhancing both inverse and surrogate predictions while reducing initial shape errors. The relatively small discrepancies shown by the colored overlays in the optimized and predicted results demonstrate that, despite the persistence of some minor imperfections, the proposed approach significantly improves accuracy relative to the initial prints. This improvement is crucial in micro- and nanoscale 3D printing, where even slight geometric variations may significantly influence performance.

The logical progression for projection multiphoton lithography is to expand our optimization methodology from planar patterns to 3D structures. Although the present Deep ResNet51-AE architecture has the potential to encode height information using higher-bit-depth input and output images, its efficacy is limited by our data-acquisition technique. Recent advancements in in-situ 3D measurement for multiphoton printing [46,47] provide a plausible solution. Integrating such volumetric measurement methods into our projection system will provide

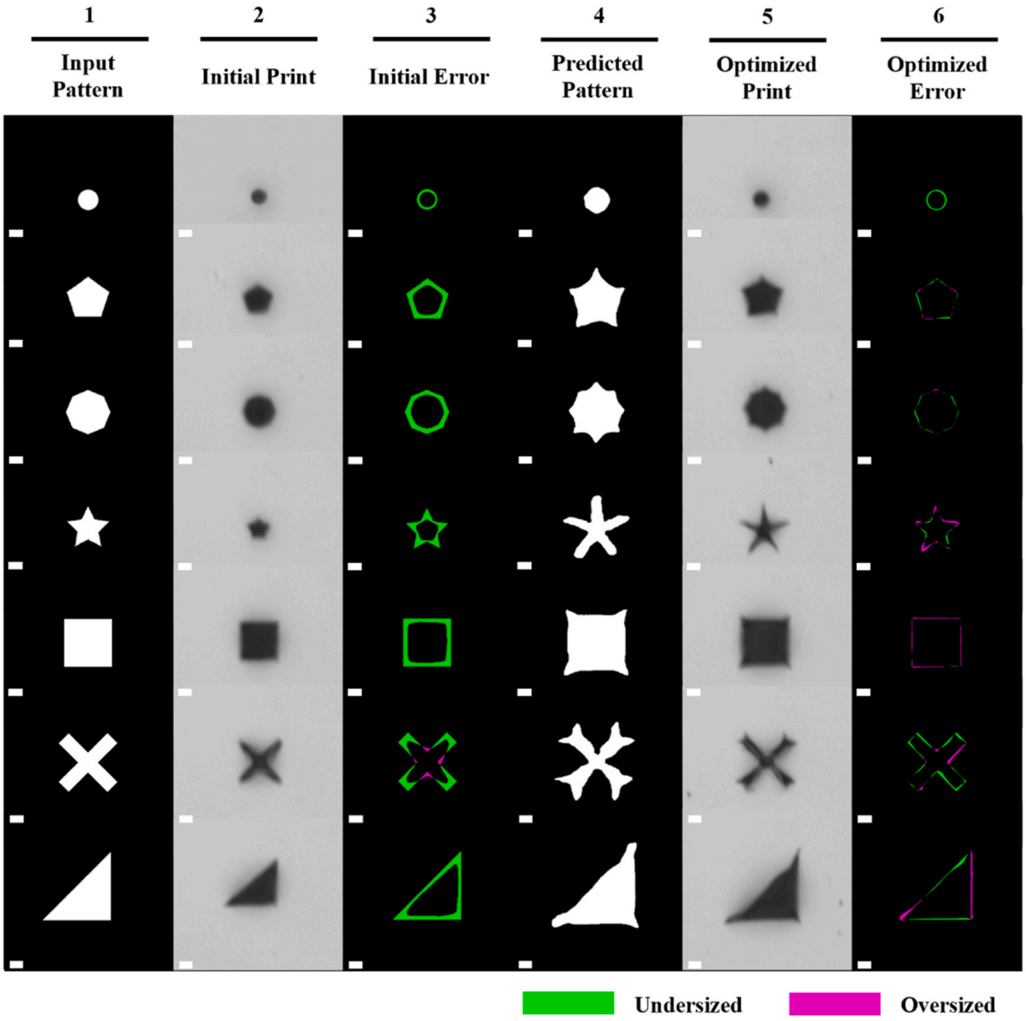

Fig. 6. Qualitative results from the Deep ResNet51-AE inverse model. Column 1 showcases some simple target patterns derived from test dataset A of various sizes that are similar to training shapes but previously unseen by the inverse model. Visualization of the corresponding printed results and the initial errors are depicted in columns 2–3. Column 4 displays the optimal patterns predicted by the inverse model. Columns 5–6 demonstrate the subsequent optimized prints and their binarizations compared with the target patterns from Column 1. The scale bars are 2 μm in length.

Table 3
Inverse model prediction perimeter error improvement on patterns taken from test dataset A.

Shape	Target Size (μm)	Perimeter Error	
		Initial (μm)	Optimized (μm)
ellipse	3.0	0.362	0.231
pentagon	6.5	0.586	0.085
octagon	6.5	0.605	0.071
star	6.5	0.346	0.122
rectangle	10.0	0.590	0.079
cross	10.0	0.388	0.161
triangle	10.0	0.496	0.188

the training data required to capture axial resolution effects and temporal exposure dynamics, therefore aiding the model in understanding layer-by-layer interactions for 2.5D-3D profile optimizations.

4. Conclusion

This study introduced a Deep ResNet51-AE framework for both surrogate and inverse modeling in projection multi-photon lithography. By systematically training on a broad spectrum of shapes and employing advanced residual learning strategies, the autoencoder can accurately predict the final printed structure (surrogate mode) and generate pre-compensated DMD patterns (inverse mode) to address dimensional inconsistencies in printed results. Experimental evaluations on complex, multi-feature test geometries demonstrated notable perimeter fidelity enhancements, reducing perimeter errors to as low as 0.07 μm for structures exceeding 4 μm in size. Even smaller shapes, which naturally exhibit higher print variability under the current setup, still benefited from the proposed approach, achieving improved dimensional accuracy compared to the conventional approach. These findings underscore the effectiveness of deep learning in elevating micro- and nanoscale additive manufacturing, as evidenced by its adaptability to local process perturbations and the ability to manage substantially more intricate designs than those used in training, where the inverse autoencoder slashed perimeter errors by a factor of three. Data-driven compensation emerges as a crucial strategy, given the profound effect that even minor geometric deviations can have on overall functionality at these scales. Furthermore, the Deep ResNet51-AE retains strong surrogate modeling capabilities, achieving equivalent predictive precision on its own inverse DMD predictions. This opens possibilities for data augmentation and simulation-based investigations aimed at unraveling the complex photopolymerization dynamics inherent to 3D multi-photon lithography. Ultimately, the Deep ResNet51-AE surrogate-inverse technique offers a promising solution to enhancing accuracy in micro- and nano-scale

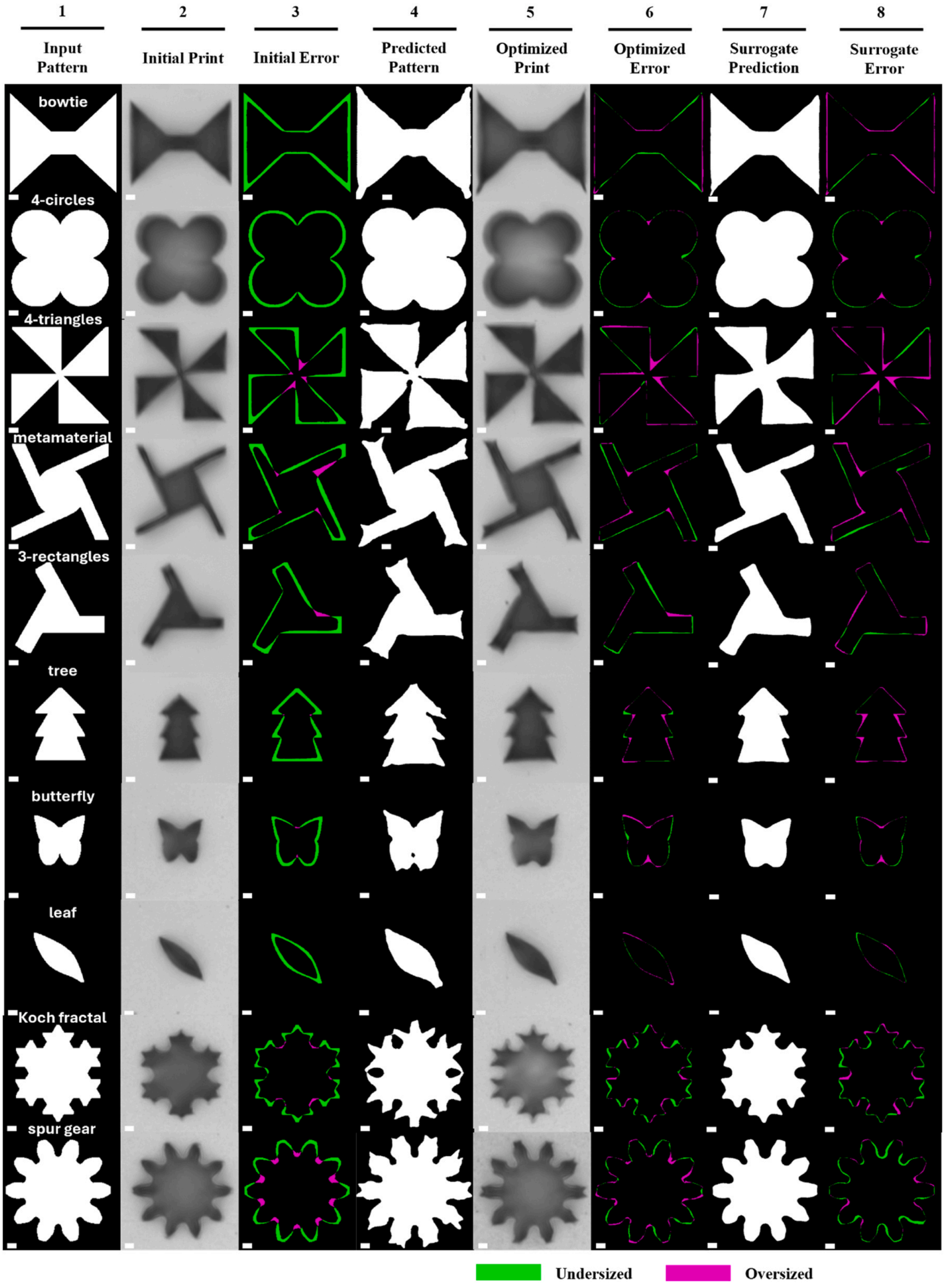

Fig. 7. Complex test shape pattern inverse and surrogate predictions by the Deep ResNet51-AE. Column 1 showcases unseen complex target patterns (derived from test datasets A and B) that are formed by compounding several simple shapes. Columns 2–3 visualize the corresponding printed results and the initial printing errors. Column 4 displays the optimal pattern predicted by the inverse model. Columns 5–6 demonstrate the subsequent optimized prints and their binarizations compared with the target patterns. Column 7 shows the surrogate model prediction on the inverse model pattern estimations, while column 8 compares the subsequent results with the target shapes. The first six target patterns are taken from the test dataset A, whereas the butterfly, leaf, Koch fractal, and spur gear patterns originate from test dataset B. The scale bars are 2 μm in length.

Table 4
Complex shape inverse and surrogate model prediction perimeter error on patterns derived from test datasets A and B.

Shape (test dataset)	Perimeter Error		
	Initial (μm)	Optimized (μm)	Surrogate (μm)
bowtie (A)	0.563	0.166	0.156
4-circles (A)	0.587	0.122	0.097
4-triangles (A)	0.482	0.171	0.195
metamaterial (A)	0.470	0.170	0.179
3-rectangles (A)	0.474	0.208	0.177
tree (A)	0.565	0.131	0.154
butterfly (B)	0.485	0.170	0.118
leaf (B)	0.481	0.075	0.078
Koch fractal (B)	0.324	0.192	0.184
spur gear (B)	0.394	0.172	0.174

additive manufacturing, overcoming key limitations in existing fabrication methods and supporting future advancements in high precision engineering.

CRediT authorship contribution statement

Ishat Raihan Jamil: Writing – review & editing, Writing – original draft, Visualization, Validation, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Jason E. Johnson:** Writing – review & editing, Methodology, Investigation, Data curation. **Xianfan Xu:** Writing – review & editing, Supervision, Project administration, Funding acquisition, Conceptualization.

Author statement

All authors agree to this submission. The work has not been published or submitted for publication elsewhere. There is no conflict of interest among authors. If accepted, the article will not be published elsewhere in the same form, in English or in any other language, including electronically, without the written consent of the copyright-holder.

Declaration of Competing Interest

There is no conflict of interest among all authors.

Acknowledgements

This research was funded by the National Science Foundation (NSF) under grant number CMMI-2135585. J.E.J. acknowledges the National Science Foundation for support under the Graduate Research Fellowship Program (GRFP) under grant number DGE-1842166. We appreciate Prof. Bryan Boudouris, Zihao Liang, and Kaushik Baruah for synthesizing the BBK photoinitiator. We would like to thank Prof. Guang Lin for providing us with the computational facilities.

Appendix A. Supporting information

Supplementary data associated with this article can be found in the online version at doi:10.1016/j.addma.2025.104929.

Data availability

The data and codes supporting the findings in this study are accessible from the corresponding authors upon reasonable request.

References

[1] T.D. Ngo, A. Kashani, G. Imbalzano, K.T.Q. Nguyen, D. Hui, Additive manufacturing (3D printing): a review of materials, methods, applications and challenges, Compos B Eng. 143 (2018) 172–196, https://doi.org/10.1016/j.compositesb.2018.02.012.

[2] R. Chaudhary, P. Fabbri, E. Leoni, F. Mazzanti, R. Akbari, C. Antonini, Additive manufacturing by digital light processing: a review, Prog. Addit. Manuf. 8 (2023) 331–351, https://doi.org/10.1007/s40964-022-00336-0.

[3] P. Somers, A. Münchinger, S. Maruo, C. Moser, X. Xu, M. Wegener, The physics of 3D printing with light, Nat. Rev. Phys. 2023 6 (2) (2023) 99–113, https://doi.org/10.1038/s42254-023-00671-3.

[4] S. Kawata, H.-B. Sun, T. Tanaka, K. Takada, Finer features for functional microdevices, Nature 412 (2001) 697–698, https://doi.org/10.1038/35089130.

[5] T. Baldacchini, Three-dimensional microfabrication by Two-Photon polymerization, in: E.A. Aránzazu del Campo (Ed.), Generating Micro- and Nanopatterns on Polymeric Materials, 1st ed., Wiley, Hoboken, NJ, 2011, pp. 107–140, https://doi.org/10.1002/9783527633449.ch7.

[6] M. Farsari, B.N. Chichkov, Two-photon fabrication, Nat. Photonics 3 (2009) 450–452, https://doi.org/10.1038/nphoton.2009.131.

[7] J. Song, C. Michas, C.S. Chen, A.E. White, M.W. Grinstaff, From simple to architecturally complex hydrogel scaffolds for cell and tissue engineering applications: opportunities presented by Two-Photon polymerization, Adv. Health Mater. 9 (2020), https://doi.org/10.1002/adhm.201901217.

[8] Z.-C. Ma, Y.-L. Zhang, B. Han, X.-Y. Hu, C.-H. Li, Q.-D. Chen, H.-B. Sun, Femtosecond laser programmed artificial musculoskeletal systems, Nat. Commun. 11 (2020) 4536, https://doi.org/10.1038/s41467-020-18117-0.

[9] Y.-L. Sun, W.-F. Dong, L.-G. Niu, T. Jiang, D.-X. Liu, L. Zhang, Y.-S. Wang, Q.-D. Chen, D.-P. Kim, H.-B. Sun, Protein-based soft micro-optics fabricated by femtosecond laser direct writing, Light Sci. Appl. 3 (2014) e129, https://doi.org/10.1038/lsa.2014.10.

[10] T. Frenzel, V. Hahn, P. Ziemke, J.L.G. Schneider, Y. Chen, P. Kiefer, P. Gumbsch, M. Wegener, Large characteristic lengths in 3D chiral elastic metamaterials, Commun. Mater. 2 (2021) 4, https://doi.org/10.1038/s43246-020-00107-w.

[11] L. Zhang, C. Wang, C. Zhang, Y. Xue, Z. Ye, L. Xu, Y. Hu, J. Li, J. Chu, D. Wu, High-Throughput Two-Photon 3D printing enabled by holographic Multi-Foci High-Speed scanning, Nano Lett. 24 (2024) 2671–2679, https://doi.org/10.1021/acs.nanolett.4c00505.

[12] A. Balena, M. Bianco, F. Pisanello, M. De Vittorio, Recent advances on high-speed and holographic two-photon direct laser writing, Adv. Funct. Mater. 33 (2023) 2211773, https://doi.org/10.1002/adfm.202211773.

[13] L. Yang, J. Li, Y. Hu, C. Zhang, Z. Lao, W. Huang, J. Chu, Projection two-photon polymerization using a spatial light modulator, Opt. Commun. 331 (2014) 82–86, https://doi.org/10.1016/j.optcom.2014.05.051.

[14] Q. Ge, Z. Li, Z. Wang, K. Kowsari, W. Zhang, X. He, J. Zhou, N.X. Fang, Projection micro stereolithography based 3D printing and its applications, Int. J. Extrem. Manuf. 2 (2020) 022004, https://doi.org/10.1088/2631-7990/ab8d9a.

[15] S.K. Saha, D. Wang, V.H. Nguyen, Y. Chang, J.S. Oakdale, S.-C. Chen, Scalable submicrometer additive manufacturing, Science 366 (2019) 105–109, https://doi.org/10.1126/science.aax8760.

[16] P. Somers, Z. Liang, J.E. Johnson, B.W. Boudouris, L. Pan, X. Xu, Rapid, continuous projection multi-photon 3D printing enabled by spatiotemporal focusing of femtosecond pulses, Light Sci. Appl. 10 (2021) 199, https://doi.org/10.1038/s41377-021-00645-z.

[17] J.E. Johnson, I.R. Jamil, L. Pan, G. Lin, X. Xu, Bayesian optimization with Gaussian-process-based active machine learning for improvement of geometric accuracy in projection multi-photon 3D printing, Light Sci. Appl. 14 (2025) 56, https://doi.org/10.1038/s41377-024-01707-8.

[18] R. Pingali, S.K. Saha, Reaction-Diffusion modeling of photopolymerization during femtosecond projection Two-Photon lithography, J. Manuf. Sci. Eng. Trans. ASME 144 (2022), https://doi.org/10.1115/1.4051830.

[19] J.E. Johnson, Y. Chen, X. Xu, Model for polymerization and self-deactivation in two-photon nanolithography, Opt. Express 30 (2022) 26824, https://doi.org/10.1364/oe.461969.

[20] J.B. Mueller, J. Fischer, F. Mayer, M. Kadic, M. Wegener, Polymerization kinetics in Three-Dimensional direct laser writing, Adv. Mater. 26 (2014) 6566–6571, https://doi.org/10.1002/adma.201402366.

[21] H. Wang, W. Zhang, D. Ladika, H. Yu, D. Gailevičius, H. Wang, C.-F. Pan, P.N. S. Nair, Y. Ke, T. Mori, J.Y.E. Chan, Q. Ruan, M. Farsari, M. Malinauskas, S. Juodkazis, M. Gu, J.K.W. Yang, Two-Photon polymerization lithography for optics and photonics: fundamentals, materials, technologies, and applications, Adv. Funct. Mater. 33 (2023) 2214211, https://doi.org/10.1002/adfm.202214211.

[22] R. Pingali, S.K. Saha, Printability prediction in projection Two-Photon lithography via machine learning based surrogate modeling of photopolymerization, J. Micro Nanomanuf. 10 (2023), https://doi.org/10.1115/1.4063021.

[23] X.Y. Lee, S.K. Saha, S. Sarkar, B. Giera, Automated detection of part quality during two-photon lithography via deep learning, Addit. Manuf. 36 (2020) 101444, https://doi.org/10.1016/j.addma.2020.101444.

[24] Y. Yang, V.A. Kelkar, H.S. Rajput, A.C. Salazar Coariti, K.C. Toussaint, C. Shao, Machine-learning-enabled geometric compliance improvement in two-photon lithography without hardware modifications, J. Manuf. Process 76 (2022) 841–849, https://doi.org/10.1016/j.jmapro.2022.02.046.

[25] D. Bank, N. Koenigstein, R. Giryes, Autoencoders, in: L. Rokach, O. Maimon, E. Shmueli (Eds.), Machine Learning for Data Science Handbook: Data Mining and Knowledge Discovery Handbook, Springer International Publishing, Cham, 2023, pp. 353–374, https://doi.org/10.1007/978-3-031-24628-9_16.

[26] O. Ronneberger, P. Fischer, T. Brox, U-net: convolutional networks for biomedical image segmentation. in: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: Eighteenth International Conference, Munich,

Germany, October 5-9, 2015, Proceedings, Part III 18, Springer, 2015, pp. 234–241.

[27] M.Z. Alom, M. Hasan, C. Yakopcic, T.M. Taha, V.K. Asari, Recurrent residual convolutional neural network based on u-net (r2u-net) for medical image segmentation, ArXiv Preprint ArXiv:1802.06955 (2018).

[28] D. Gong, L. Liu, V. Le, B. Saha, M.R. Mansour, S. Venkatesh, A. van den Hengel, Memorizing normality to detect anomaly: memory-augmented deep autoencoder for unsupervised anomaly detection. in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 1705–1714.

[29] S. Chattopadhyay, A.C. Kak, Uncertainty, edge, and REverse-attention Guided Generative adversarial network for automatic building detection in remotely sensed images, IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens 15 (2022) 3146–3167, https://doi.org/10.1109/JSTARS.2022.3166929.

[30] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition. in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[31] P. Kiefer, V. Hahn, M. Nardi, L. Yang, E. Blasco, C. Barner-Kowollik, M. Wegener, Sensitive photoresists for rapid multiphoton 3D laser Micro- and nanoprinting, Adv. Opt. Mater. 8 (2020) 2000895, https://doi.org/10.1002/adom.202000895.

[32] G. Bradski, The opencv library, Dr. Dobb`S. J. Softw. Tools Prof. Program. 25 (2000) 120–123.

[33] N. Otsu, A threshold selection method from gray-level histograms, Automatica 11 (1975) 23–27.

[34] A.S. Jariwala, F. Ding, A. Boddapati, V. Breedveld, M.A. Grover, C.L. Henderson, D. W. Rosen, Modeling effects of oxygen inhibition in mask-based stereolithography, Rapid Prototyp. J. 17 (2011) 168–175, https://doi.org/10.1108/13552541111124734.

[35] J.R. Tumbleston, D. Shirvanyants, N. Ermoshkin, R. Janusziewicz, A.R. Johnson, D. Kelly, K. Chen, R. Pinschmidt, J.P. Rolland, A. Ermoshkin, E.T. Samulski, J. M. DeSimone, Continuous liquid interface production of 3D objects, Science 347 (2015) 1349–1352, https://doi.org/10.1126/science.aaa2397.

[36] C. Arnoux, L.A. Pérez-Covarrubias, A. Khaldi, Q. Carlier, P.L. Baldeck, K. Heggarty, A. Banyasz, C. Monnereau, Understanding and overcoming proximity effects in multi-spot two-photon direct laser writing, Addit. Manuf. 49 (2022) 102491, https://doi.org/10.1016/j.addma.2021.102491.

[37] I.M. Sobol, On the distribution of points in a cube and the approximate evaluation of integrals, USSR Comput. Maths. Math. Phys. 7 (1967) 86–112, https://doi.org/10.1016/0041-5553(67)90144-9.

[38] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, A. Lerer, Automatic Differentiation in Pytorch (2017).

[39] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks. in: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, JMLR Workshop and Conference Proceedings, 2010, pp. 249–256.

[40] S. Khan, H. Rahmani, S.A.A. Shah, M. Bennamoun. A Guide to Convolutional Neural Networks for Computer Vision, first ed., Springer International Publishing, Cham, 2018 https://doi.org/10.1007/978-3-031-01821-3.

[41] L.-F. Dong, Y.-Z. Gan, X.-L. Mao, Y.-B. Yang, C. Shen, Learning deep representations using convolutional Auto-Encoders with symmetric skip connections, in: in: Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2018, pp. 3006–3010, https://doi.org/10.1109/ICASSP.2018.8462085.

[42] X. Huang, S. Belongie, Arbitrary style transfer in real-time with adaptive instance normalization. in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 1501–1510.

[43] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, Commun. ACM 60 (2017) 84–90.

[44] O. Rainio, J. Teuho, R. Klén, Evaluation metrics and statistical tests for machine learning, Sci. Rep. 14 (2024) 6086, https://doi.org/10.1038/s41598-024-56706-x.

[45] V. Sovrasov, ptflops: a flops counting tool for neural networks in pytorch framework, (n.d.). ⟨https://github.com/sovrasov/flops-counter.pytorch⟩ (accessed March 24, 2025).

[46] R. Zvagelsky, P. Kiefer, J. Weinacker, M. Wegener, In-situ quantitative phase imaging during Multi-photon laser printing, ACS Photonics 10 (2023) 2901–2908, https://doi.org/10.1021/acsphotonics.3c00625.

[47] R. Zvagelsky, F. Mayer, D. Beutel, C. Rockstuhl, G. Gomard, M. Wegener, Towards in-situ diagnostics of multi-photon 3D laser printing using optical coherence tomography, Light. Adv. Manuf. 3 (2022) 466–480, https://doi.org/10.37188/lam.2022.039.