Closed-Loop Control of Direct Ink Writing via Reinforcement Learning

Supplementary Material

MICHAL PIOVARČI*, ISTA, Austria

MICHAEL FOSHEY*, MIT CSAIL, USA

JIE XU, MIT CSAIL, USA

TIMMOTHY ERPS, MIT CSAIL, USA

VAHID BABAEI, MPI Informatics, Germany

PIOTR DIDYK, Università della Svizzera italiana, Switzerland

SZYMON RUSINKIEWICZ, Princeton University, USA

WOJCIECH MATUSIK, MIT CSAIL, USA

BERND BICKEL, ISTA, Austria

1 INTRODUCTION

The supplementary material describes additional details about the printing hardware (Section 2), numerical simulation (Section 3), and reinforcement learning formulation (Section 4). We also include the full physical results manufactured for this paper, (Section 5).

2 HARDWARE APPARATUS

We developed a direct write 3D printing platform with an optical feedback system that can measure the dispensed material real-time, in-situ. The 3D printer is comprised of a pressure-driven syringe pump and pressure controller (Enfield Technologies), a 3-axis Cartesian robot (Hiwin KK), an optical imaging system, a back-lit build platform, 3D-printer controller, and CPU, (Figure 1). The 3-axis Cartesian robot is used to locate the build platform in the x and y-direction and the print carriage in the z-direction. The pressure-driven syringe pump and pressure controller are used to dispense an optically translucent material onto the back-lit build platform.

The back-lit platform is used to illuminate the dispensed material. The movement of the robot, actuation of the syringe pump, and timing of the cameras are controlled via the controller. The CPU is used to process the images after they are acquired and compute updated commands to send to the controller.
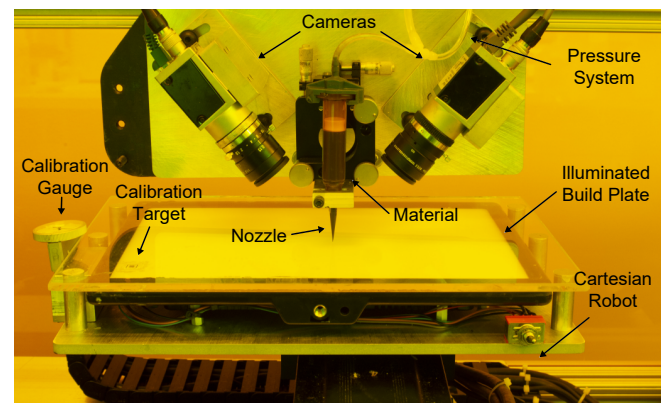

Fig. 1. The printing apparatus consisting of a 3-axis Cartesian robot, a direct write printing head, a camera setup, illuminated build plate, calibration facilities, and a orange UV-cover.

2.1 Vision Module

To enable realtime control of the printing process, we implemented an in-situ view of the material deposition. Due to the occlusions caused by the dispensing nozzle, no single camera can capture the full view. Therefore, we opted for a two-camera setup. More specifically, we place two CMOS cameras (Basler AG, Ahrensburg, Germany) at 45 degrees on each side of the dispensing nozzle, (Figure 1). We further process the images from the cameras to obtain a single top-down view of the deposition. We start by calibrating the camera by collecting a set of images and estimating its intrinsic parameters, (Figure 2 calibration). To obtain a single top-down view, we capture a calibration target aligned with the image frames of both cameras, (Figure 2 homography). We can stitch the images into a single view from an over-the-top virtual camera by calculating the homography

---

Submission ID: 528. 2022-05-12 12:04. Page 1 of 1–5.

ACM Trans. Graph., Vol. 41, No. 4, Article 112. Publication date: July 2022.

between the captured targets and an ideal top-down view. Finally, we mask the location of each nozzle in the image (Figure 2 nozzle masks) and obtain the final in-situ view, (Figure 2 stitched image).
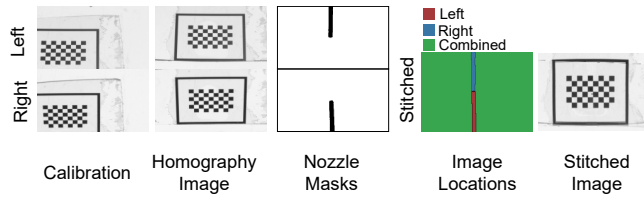

Fig. 2. The calibration of the imaging setup. First intrinsic parameters are estimated from calibration patterns. Next, we compute the extrinsic calibration by calculating homographies between the cameras and an overhead view. We extract the masks by thresholding a photo of the nozzle. The final stitched image consists of 4 regions: (1) view only in the left camera, (2) view only in the right camera, (3) view in both cameras, (4) view in no camera. The final stitched image is shown on the right.

To extract the thickness of the deposited material, we rely on its translucency properties. More precisely, we correlate the material thickness with its optical intensity, (Figure 3). We do this by depositing the material at various thicknesses and taking a picture with our camera setup. The optical intensity then decays exponentially with an increased thickness.
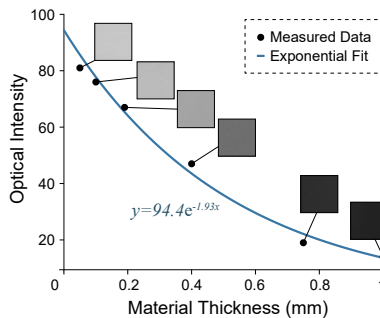

$$y = 94.4e^{-1.93x}$$

Fig. 3. Calibration images for correlating deposited material thickness with optical intensity and the corresponding fit.

2.2 Mechanical Calibration

In our simulation, we assume the needle is centered with respect to the in-situ view. To ensure that this assumption holds with the physical hardware, we calibrate the location of the dispensing needle within the field of view of each camera and with respect to the build platform. First, a dial indicator is used to measure the height of the nozzle in z, and the fine adjustment stage is adjusted until the nozzle is 254 microns above the print platform. Next, using a calibration target located on the build platform and the fine adjustment stage, the nozzle is centered in the field of view of each camera. This calibration procedure is done each time the nozzle is replaced during the start of each printing session.

3 SIMULATION DETAILS

The discretization choice of the numerical model affects the learning process. We experimented with two options: (1) time-based and (2) distance-based. We originally experimented with time-based discretization. However, we found out that time discretization is not suitable for printer modeling. As the velocity in simulation approaches zero, the difference in deposited material becomes progressively smaller until the gradient information completely vanishes, (Figure 4 top). Moreover, a time-based discretization allows the policy to affect the number of evaluations of the environment directly. As a result, it can avoid being punished for bad material deposition by quickly rushing the environment to finish. Considering these factors we opted for distance-based discretization, (Figure 4 bottom). The policy specifies the desired velocity at each interaction point, and the environment travels a predefined distance (0.315 mm) at the desired speed. This helps to regularize the reward function and enables learning of varying control policies.
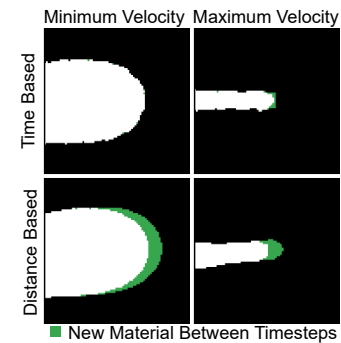

Fig. 4. The amount of material deposited between two simulation steps depends on the discretization choice. With time-based discretization, the reward gradient vanishes at low speeds. In contrast, distance-based discretization produces a more uniform response.

An interesting design element is the orientation of the control polygons created by the slicer. When the outline is defined as points given counter-clockwise, then due to the applied rotation, each view is split roughly into two half-spaces, (Figure 5). The bottom one corresponds to outside i.e., generally black, and the upper one corresponds to inside i.e., generally white. However, the situation changes when outlining a hole. When printing a hole the two half-spaces swap location. We can remove this disambiguity by changing the orientation of the polylines defining holes in the model. By orienting them clockwise, we will effectively swap the two half-spaces to the same orientation as when printing the outer part. As a result, we achieve a better usage of trajectories and a more robust control scheme that does not need to be separately trained for each print's outer and inner parts.

4 REINFORCEMENT LEARNING FRAMEWORK

To train our control policy, we start with a g-code generated by a slicer. As inputs to the slicer, we consider a set of 3D models collected from the Thingy10k dataset. To train a controller, the input models need to be carefully selected. On the one hand, if we
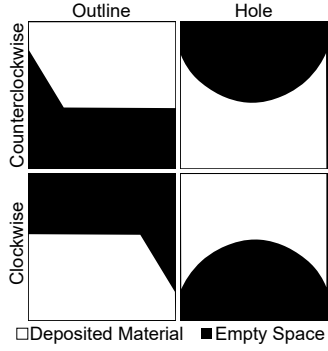
Fig. 5. The observation space is split into two half-spaces corresponding to the target printout and void. Therefore, it is advantageous to pick different printing directions for objects' outlines and holes inside to minimize the observation space and reuse trajectories. By picking different directions, we guarantee that material is always deposited only on one side.

pick an object with too low-frequency features with respect to the printing nozzle size, then any printing errors due to control policy will have negligible influence on the final result. On the other hand, if we pick a model with too high-frequency features regarding the printing nozzle, the nozzle will be physically unable to reproduce these features. As a result, we opted for a manual selection of 18 models that span a wide variety of features, (Figure 6). Each model is scaled to fit into a printing volume of 22×22 mm and sliced at random locations.

Fig. 6. Models contained in our training curriculum.

Our policy is represented as a CNN modeled after Mnih et al. [2015]. The network input is a $84 \times 84 \times 3$ image. The image is passed through three hidden layers. The convolution layers have the respective parameters: (32 filters, filter size 8, stride 4), (64 filters, filter size 4, stride 2), and (64 filters, filter size 3, stride 1). The final convolved image is linearized and passed through a fully-connected layer with 512 neurons connected to the output actions. Each hidden layer uses the nonlinear rectifier activation. We formulate our objective function as in [Schulman et al. 2017]:

$$\arg\max_{\theta} \mathbb{E}_t^C \left[\frac{\pi_{\theta_t}(a_t|s_t)}{\pi_{\theta_{t-1}}(a_t|s_t)} \hat{A}_t \right], \quad (1)$$

where t is a timestep in the optimization, θ are the hyperparameters of a neural network encoding our policy π that generates an action a_t based on a set of observations s_t, $\hat{A}_t$ is the estimator of the advantage function and the expectation $\mathbb{E}_t^C$ is an average of a finite batch of samples generated by printing sliced models from our

curriculum C. To maximize Equation 1 we use Principal Policy Optimization (PPO) algorithm [Schulman et al. 2017].

4.1 Learning Curves

We conducted several ablation studies during which we observed the convergence of our learning process. When experimenting with the observation space, we did not observe a significant difference in learning convergence between our full observation space and its reduced variants, (Figure 7). During our experiments with action space, we found that a policy that optimizes only the velocity of the nozzle can learn faster than a policy that adjusts the deposition path, (Figure 8 left). Lastly, we observed the convergence rate between different reward computation strategies, (Figure 8 right). The delayed reward converges significantly slower than a reward function with instantaneous feedback, and it is unclear if a performance similar to our privileged reward can be achieved.
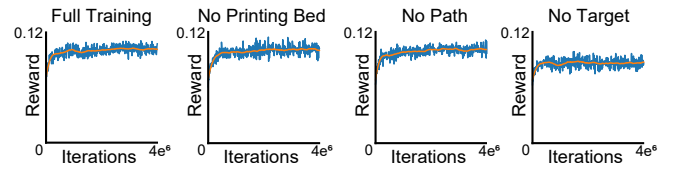

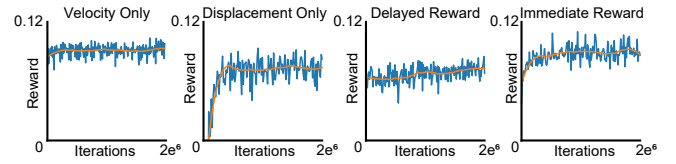Fig. 7. Training curves for controllers with constant material flow.

Fig. 8. Training curves for controllers with variable material flow.

5 RESULTS

For evaluation, we constructed a dataset consisting of freeform and CAD geometries that were not present in training. A subset of the dataset is visualized in Figure 9.

Fig. 9. Exemplar models from the evaluation dataset.

5.1 Slices Used to Estimate Outline Improvement

We estimate the quality of deposition by evaluating the under and over deposition histograms on a subset of the evaluation dataset, (Figure 10).
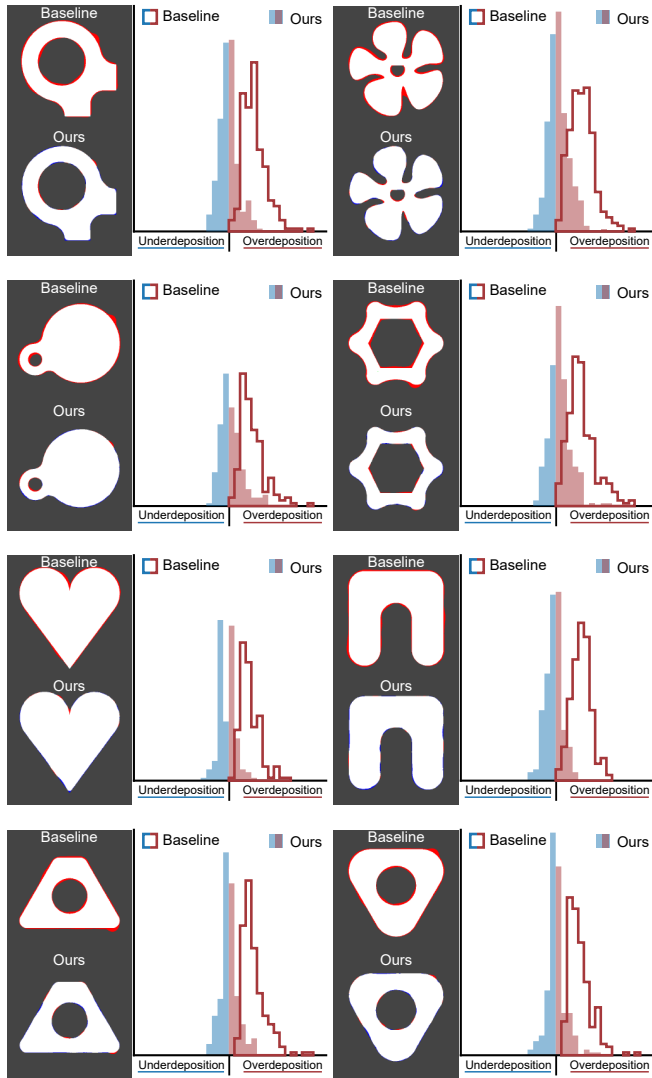
Fig. 10. Recovered histograms for a subset of slices from the evaluation dataset show tighter deposition of material achieved by our closed-loop control policy.

5.2 Detailed Physical Results

We fabricated 11 shapes from the training and 11 shapes from the evaluation dataset on a physical apparatus using a low and high viscosity material, (Figure 11).

REFERENCES

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. 2015. Human-level control through deep reinforcement learning. *Nature* 518, 7540 (2015), 529–533.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).

ACM Trans. Graph., Vol. 41, No. 4, Article 112. Publication date: July 2022.
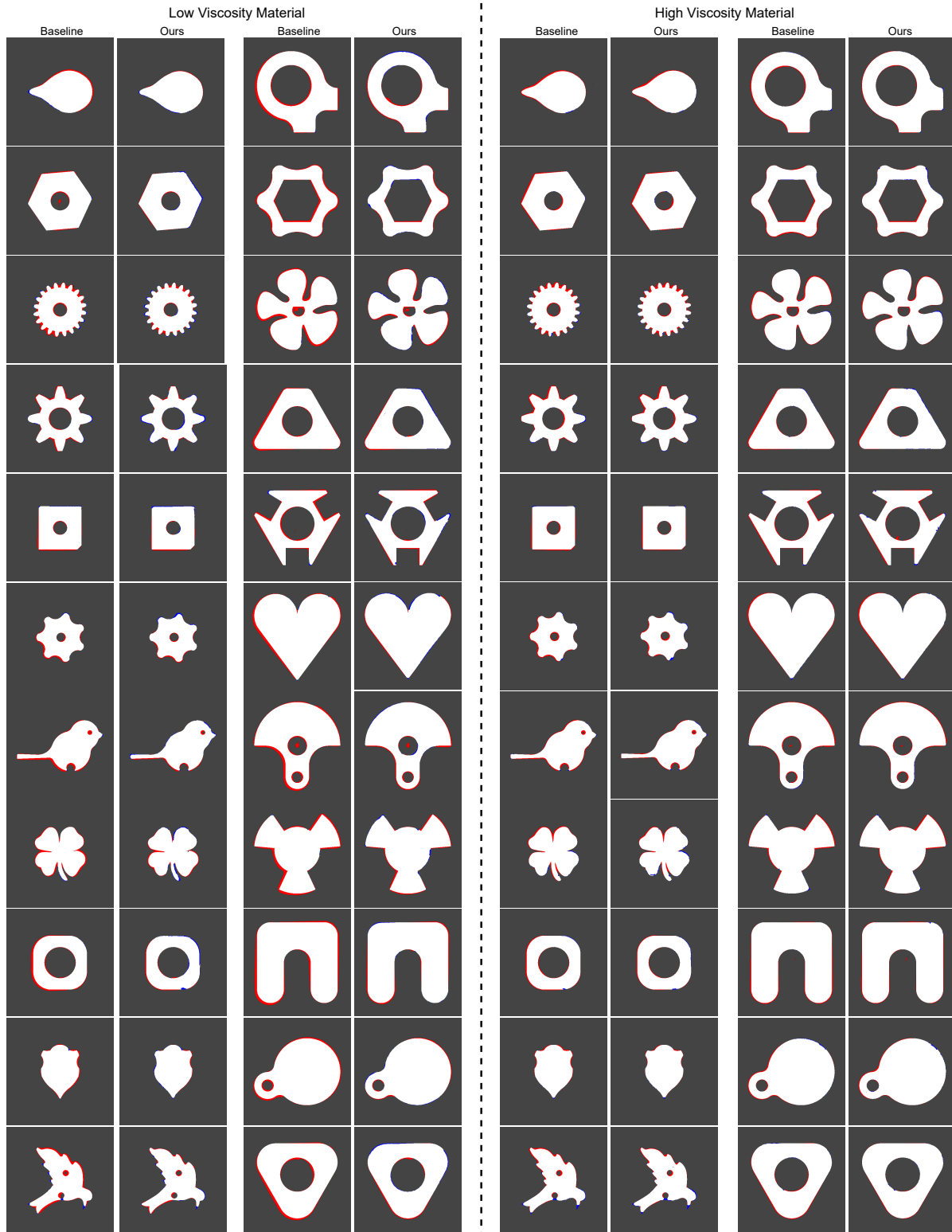
Submission ID: 528. 2022-05-12 12:04. Page 4 of 1–5.

Fig. 11. Policy evaluation on physical hardware.