ConTextural: A Toolpath-Based Texture Editing Tool for Extrusion 3D Printers

Daphna Kaplan
daphnakaplan@campus.technion.ac.il
Technion Institute of Technology
Haifa, Israel

Mirela Ben Chen
mirelab@gmail.com
Technion Institute of Technology
Haifa, Israel

Yoav Sterman
sterman.yoav@technion.ac.il
Technion Institute of Technology
Haifa, Israel

a) download b) paint c) visualize d) print

Figure 1: An overview of ConTextural workflow. a) Download 3D model. b) Use the tool to paint the model with the desired textures. c) View the visualization to check the expected results. d) Generate Gcode and print.

ABSTRACT

Recently, there has been increased interest in design tools for creating textures using toolpath manipulation for extrusion-based 3D printers. Most tools are limited in their ability to edit existing 3D models and the variety of possible textures. Here, we present ConTextural, a design tool for adding texture to existing 3D models using toolpath manipulation. Using a coloring-based user interface, ConTextural allows users to draw textures on 3D models. Inspired by knitting structures, we introduce the concept of *texture primitives*, constructing *texture structures* that enable abundant possibilities for texture patterns. We include a curated texture library, enabling users to easily craft intricate and personalized designs. We assess the tool's impact on users' expressiveness, engagement, and satisfaction using a user study and demonstrate how it helps to produce uniquely distinct designs from a single 3D model. Additionally, we provide design examples highlighting functional applications for adding textures to existing 3D models.

CCS CONCEPTS

• **Human-centered computing → Interactive systems and tools**.

KEYWORDS

Design workflows, Creativity support tools, Creative expression, Unit structures, Fabrication

ACM Reference Format:
Daphna Kaplan, Mirela Ben Chen, and Yoav Sterman. 2025. ConTextural: A Toolpath-Based Texture Editing Tool for Extrusion 3D Printers. In *Proceedings of ACM Conference (Conference'17)*. ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Textures play a crucial role in design, affecting both the aesthetic and functional qualities of an object. Visually, textures are a key element, adding complexity and depth to surfaces and allowing expressiveness in the design. Tactilely, textures influence the way an object feels to the touch as explained by Szczesniak et al. [55]. Tactile surfaces are used in product design and in creating tangible interfaces. Textures can also add functional properties to an object. For example, adding friction can improve the grip of handles as seen in Pawlus et al. [40].

Usually, textures are defined during the design phase of the model using Computer-Aided Design (CAD) software. However, extrusion-based 3D printing provides the opportunity to generate intricate textures by manipulating the *printing toolpath* – the path the printer head follows – and the printing parameters associated with it. The toolpath is converted into a printing procedure through the Geometric Code (Gcode) file format. The Gcode file is usually generated by a slicer software, which slices the 3D model into layers and calculates the toolpath for each layer. Slicers are designed to ensure printability and accurate translation of the 3D model into physical space. This process is technical and does not involve creative interactions. Commercial slicers do not enable texture creation using toolpath manipulation. However, some slicers, such as Cura and Prusa [62], have a feature called "Fuzzy Skin", which randomly distorts the toolpath on each layer. This feature is applied globally to the entire model and generates only one "flavor" of texture.

There are several advantages to manipulating the toolpath directly: First, while slicers define *global* print settings, a custom tool can provide finer control over each segment along the path and its parameters, such as defining a custom extrusion rate at a specific point. Second, it can harness both physical (e.g., extrusion amount) and environmental factors (e.g., gravity), highlighting the richness of the material. Examples include the creation of hair-like bristles by Laput et al. [15], wavy surfaces by Takahashi et al. [57], entangled threads by Lipton et al. [30] and oozing effects by Cohen et al. [9]. These toolpath texture geometries are fine-detailed, intricate, and often unpredictable, making them difficult, and sometimes impossible, to model and fabricate using CAD software and conventional slicing.

The interest in the design possibilities of toolpath textures, combined with the lack of commercial interactive design tools for generating them, has led researchers and makers to develop custom slicers. Developing such tools requires expertise in programming, machine code, hardware, and materials. Some of the recent examples of custom slicers include WeaveSlicer [13], which improves the printability of clay materials by adding a weaving texture to the toolpath while also improving the aesthetic value. ClayToolkit is a framework by Toka et al. [59] in Grasshopper, allowing parametric pattern definition and local editing of textures. Another example is CoilCAM [6], a CAM programming system, which focuses on the creative aspects of clay printing. CoilCAM generates the printing path using geometric and mathematical stackable functions to generate a variety of shapes and surface textures. While these examples enable toolpath manipulation, they are based on the user's ability to work within a coding environment and therefore are not accessible for 3D printing users without coding knowledge.

Current toolpath editing tools have several limitations. First, many of these tools create 3D models by generating toolpath data, but they do not support importing pre-existing 3D models, limiting their option for model integration. Second, these tools often focus on a single texture type or provide only a narrow range of texture options, which restricts creative control over the appearance of the model. Finally, most of these tools do not include an accessible user interface for locally editing textures, making it challenging for users to modify textures directly on specific areas of the model.

To address the need for a simple and accessible tool for applying toolpath-based textures to 3D models, we present **ConTextural**: *A Toolpath-Based Texture Editing tool for Extrusion 3D Printers.* This tool combines the advantages of direct toolpath interaction with the ability to import existing 3D models from online libraries or CAD software. ConTextural enables users to engage in a design process, facilitating a material-focused and expressive approach to 3D printing. The tool allows users to apply textures to 3D models using a paintbrush, an editing process that is common in 3D modeling, and 2D design interfaces. With the brush, users can define and edit multiple areas with different textures, and use the visualization mode to get better insight into the expected printed results. The tool features an accessible user interface to users without previous coding or gcode knowledge. The Gcode for printing the model is generated automatically.

Our approach to defining toolpath-based textures is inspired by concepts from *digital knitting*. While knitting and 3D printing differ in many ways, they share key similarities: both are additive processes that build layer after layer, manipulating a single continuous material. Knitting has 4 basic operations (knit, tuck, float, and transfer) [48]. When combined into knitting structures, which are matrices of operations, these limited number of operations generate a large variety of patterns and designs, as shown in KnitPicking Textures [23]. In a similar vein, we define 6 basic *texture primitives*: None, Thick, Arc, Wiggle, Dot, and Hair, whose toolpaths differ significantly from their final geometry due to material behavior under gravity and printing parameters. The primitives are based on methods that were previously developed in the field, but by combining them into *structures* we create a new and expansive design space of textures.

Our main contributions are:

(1) An abstraction of toolpath manipulation for texture generation using a modular system of structures, each composed of a set of primitives.
(2) A coloring-based tool that supports a modeling-free workflow for editing existing models, enabling personalization and engagement for 3D printing users.
(3) A library that contains tested, ready-to-use *texture structures* that create a rich design space.

We conducted a user study with ten participants of varying levels of expertise in 3D printing, modeling, and coding to evaluate the tool's impact on their creative experience. We find that most users are highly satisfied with the results and report that the results were worth the effort. We present the participants' designs and discuss the results. Additionally, we provide examples showcasing how textures can be applied to existing 3D models for both functional and aesthetic purposes, demonstrating the potential of the design space.

2 RELATED WORK

In this section, we review existing approaches from the literature on three fields. First, we cover the motivations related to texture generation in 3D printing. Then we review research on tools focusing on texture generation through CAD and methods based on Computer-Aided Manufacturing (CAM) operations. Lastly, we touch upon studies where a units-based design system provides different levels of abstraction.

2.1 Textures in 3D Printing

There are several motivations for incorporating textures into 3D-printed artifacts. Textures play a key role in the aesthetics of a design, making them a desired feature for artists and designers, as noted by Ashby and Johnson [5]. In clay printing, where material appearance is a primary focus, artists often explore textures as a central element of their work. This exploration is exemplified in the research of Gursoy et al. [19], which highlights the exploration process in ceramic printing.

Beyond aesthetics, textures can also alter the physical properties of an artifact. For instance, in 4D printing, textures can enable transformable surfaces, as demonstrated by Sun et al. [53]. Textures also contribute to the haptic qualities of tangible interfaces, such as the vibratory textures used in Surface I/O by Ding et al. [10]. Additionally, textured designs find applications in fields like food printing, where texture plays a critical role in improving the product experience, as reviewed in Pereira et al. [41]. Creating a texture for 3D printing can be done through Computer-Aided Design (CAD) tools or Computer-Aided Manufacturing (CAM) tools.

2.2 Adding textures using CAD

Computer-Aided Design (CAD) software like SolidWorks and Rhino allows modeling the geometry of the model and its texture. However, this can be a complex task, therefore several approaches exist to assist in automating the process. Parametric design tools such as Grasshopper [44] and Blender allow users to create parametric definitions. Additionally, Algorithmic methods have been developed, often using external input to generate and modify textures more efficiently.

One common input for generating textures is a 2D image used as a height map, mapped onto the 3D model's surface, for instance, in Verma et al. [64]. Similarly, HapticPrint [61] allows adding a desired feel to objects using a 2D vector image through a user interface. The textures are based on a grid pattern of cells, which the user can define using an image of proper format. HapticPrint also includes a library of printable reference designs, and an internal pattern design as well. However, the design space enabled by a height map is limited to an embossment of the surface. Additionally, HapticPrint does not discuss local or partial textures.

Another interface with a 2D input is presented in Tabby [54] which allows users to import a drawing and automatically apply it as a repeating pattern for making an embossed pattern. With Tabby, the user can apply the texture to a selected semantic region of the model. Similar to height maps, the design space of textures is limited to solid embossing on the surface. Moreover, regional area selection is done automatically.

While CAD tools focus on modeling the geometric shape of the model and its surface, they do not interact with, or control the printing process itself. Hence, they don't leverage the advantages of a toolpath to control the textural appearances. Additionally, they don't take advantage of toolpath manipulation to affect the physical properties of the model such as the strength, roughness, and more.

2.3 Textures through CAM

When generating the toolpath using CAM-based tools, the model's shape is defined using the toolpath route, including the texture. Defining the shape using the toolpath reveals a different design space than CAD-based tools do. Therefore, ceramic printer manufacturers, like WASP [65], develop apps for customizing vase-like contours with a set of sliders. Similarly, PotterWare [45] allows similar interaction, including external files; however, neither tool provides local adaptations, custom definitions, or visualization of the final product.

Additional research into generating the shape through the toolpath suggests different interaction methods. PotScript [34] is a web app with a drag-and-drop interface to generate a pot's Gcode without writing any code. To increase users' insight into the printing process details, FullControl [17] introduces an Excel-based control to set the desired properties as suggested in their format. Conceptualizing the toolpath as mathematical actions is done by CoilCAM [6] for producing clay printing toolpaths. Other tools focus on exploring new interaction mediums. An augmented reality interface for editing the toolpath by controlling different modifiers is presented by Passananti et al. [38]. Most recently, SketchPath presented a drawing interface that allows drawing and editing the toolpath using a digital sketching board [14].

Other tools focus on real-time user intervention in the fabrication process through physical gestures: Kim et al.[27] provides physical inputs for interacting with the process such as sketches, objects, and more, and Fossdal et al. [11] developed rich interactions for controlling digital fabrication machines' toolpaths and parameters. In Goudswaard et al. [18] users can manually manipulate a printer axis to become more embedded in the process while exploring the resulting effect.

Custom CAM-based methods allow for the design of textures by interaction with the machine code and the printing parameters. A custom slicer is developed by Yan et al. [66] to create a more natural-looking texture appearance using an image, such as wood or tiles. A mapping is made between the texture and toolpath layers to apply a geometric transformation. Another example of image-based input for interacting with the toolpath is presented in Velocity Painting [33], allowing users to project an image onto a model, affecting the print speed, which creates an appearance variation. Image inputs are an effective method for applying texture, however, they are applied globally, leaving out local adjustments. In addition, they might create warping deformations due to the mapping process.

Focusing on the toolpath geometry and its parameters allows the exploration of different textures and structures. Takahashi and Miyashita [58] created textures by manipulating parameters like the z-offset and extrusion amount. Similarly, in a different project, extrusion rates and amounts are used to create tactile sheets [57] . Laput et al. [16] exploits the material stretchability to create hair-like covered surfaces. O'dowd et al. [37] explores 'expressive modes of fabrication', presenting a 'plucked texture', lace texture, and more. In Mohite et al. [35] the deposition speed is varied to explore different textures in clay. These works focus on the method of texture production rather than the method of applying texture. Their main contribution is the ability itself, and they do not provide a user interface.

Due to the lack of out-of-the-box options that provide versatility in toolpath design, an important development was made in publishing dedicated coding libraries for toolpath control. Extruder-Turtle is a Python library for laying out the path the printer makes, which corresponds to the commands written to the Gcode file. The library allows recursive drawing definitions, and other patterns of design [42]. Similarly, P5.fab [52] is a system for controlling 3D printers through the creative coding environment P5. 'P5.fab' allows the exploration of print parameters through greater control over the fabrication processes. Nevertheless, these tools require coding knowledge and Gcode expertise even for simple models and are therefore not accessible to casual users.

To allow users to add textures onto models without coding, Marciniak et al. [32] presented 'Texture-Slicer', an interface for applying a texture, based on a geometric manipulation of the toolpath

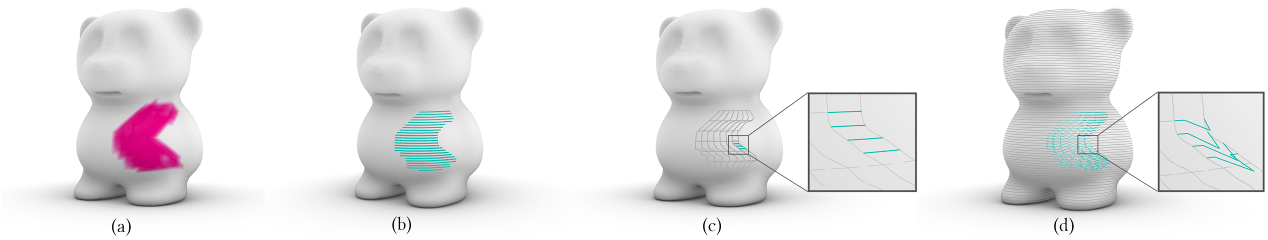


Figure 2: Generating the toolpath: a) The input model is colored by the user. b) Iso-z curves for the selected region. c) Partition to unit structures. d) Segment modification according to the primitives in each unit.

by adding "waves" with controllable amplitude and wavelength. This is a single-style slicer. We continue the trend of no-coding user interfaces for texture application through toolpath manipulation and present an approach for thinking of textures as versatile modular units. Additionally, we expand the capabilities of current tools by providing access to local modifications onto existing models.

2.4 Unit-based design systems

Design tools that handle large design spaces often require a level of abstraction due to the complexity and the amount of information. This abstraction can be done by dividing the space into a set of discrete units. This approach can be used to manage fabrication processes as well. For instance, in the realm of metamaterials, structures are crafted from cell units, each corresponding to a shape-shifting transformation. The unit's composition creates the overall movement effect, as demonstrated in Ion et al. [24]. Similarly, the work presented by Amorim et al. [4] uses flexible materials to fashion a desired shoe sole while controlling its flexibility characteristics.

In the domain of textiles, the necessity to deconstruct designs into smaller units is clear, since the number of possibilities is vast, as facilitated by digital knitting and weaving software. Sterman and Almog at el. [49] introduces knit-structure tiles to represent various stitches, forming color gradients and accommodating different structure sizes, similar to dithering in graphic art. Additionally, Albaugh et al. [2] has developed a tool enabling the use of brushes of varying sizes to select areas within the weaving pattern, thereby simplifying access to patterns from a defined pattern repository. Yu et al. [68] allows both high and low-level knitting code structures to control a knitting process.

3 CONCEPT AND MOTIVATIONS

We present a design tool aiming to enhance the user's experience through enabling expressivity and engagement by assigning textures with toolpath manipulation and Gcode generation. Existing research has explored the generation of custom Gcode to achieve specific textures, yet toolpath editing is hard to do, even for users with experience in 3D printing and coding . We propose a simple workflow for accomplishing this objective: *download* → *edit* → *visualize* → *print* (Figure 1). This workflow encourages users to engage with ready-made models by assigning textures and infusing their designs to create original and personalized designs.

Our concept for constructing texture is inspired by textile design. We make an analogy between digital knitting structures and toolpath manipulation operations. The extrusion coil can be seen as analogous to the knitting thread - a continuous line that, at each segment, goes through a set of needle actions. In most knitting software, the design is described by a color-coded bitmap. The user defines the needle actions that manipulate the yarn at each pixel: knit, tuck, float, or transfer. Combining these needle actions can produce a large variety of knitting patterns and textures. Since a program might include thousands of pixels, defining the operation for each needle action would be too tedious. Therefore, the design is broken into discrete units called knitting structures, which are matrices of needle actions. Knitting design software allows users to replicate structures across the design pattern and build new structures by combining needle actions.

Similarly, we define texture *primitives* and combine them to build *structures* for texturing 3D models. We present a library of structures, which are examples of the possible combinations enabled by our approach. This concept is extendable and modular to support the future expansion of the texture structures library. Each structure consists of a set of *primitives* applied to the relevant toolpath segments (See 4.1). We use color coding to associate the selected textures in the user interface to the selected locations painted on the model (see Fig. 6). Using a coloring brush, the user applies the different structures to the model.

To build our tool, we provide a high-level abstraction for the users while performing the low-level operations through the software. The tool is domain-specific for its goal and is made to be extendable with the option of exposing more low-level operations to allow different levels of abstraction. We believe this follows the suggestions provided by Li et al. [29] for building empowering creative support tools.

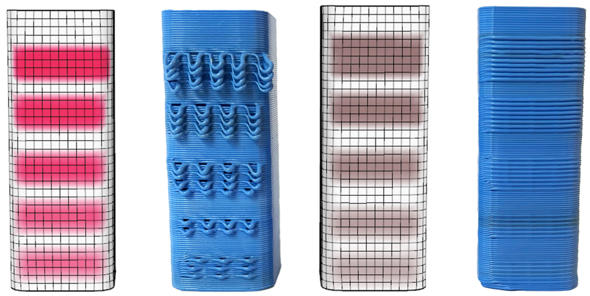
4 SYSTEM OVERVIEW

We developed our prototype inside the "Grasshopper for Rhino" environment [44]. The user interface is implemented using the HumanUI plugin [21]. We used a PollenAM PAM pellets 3D printer [43] for our fabrication setup with a nozzle size of 0.8mm.

4.1 Terminology

We define three main terms:

- *Segment*: a single curve in a structure, which is a small subcurve of an original iso-z curve of the model.



(a) (b)

Figure 3: The mapping between the color opacity (left) and the corresponding texture intensity (right) for two examples: loose arcs (a), and partially thick (b).

- *Texture Primitive*: a definition of a geometric and printing parameter manipulation on a segment.
- *Texture Structure*: an ordered list of paired segments and primitives that together form the texture.

Texture Primitives. Primitives are types of curves, defined by the way they manipulate the toolpath. We define a manipulation as a *geometric* change, a *parameter* change, or both. Additionally, each primitive has an *intensity* property, which can create a minor or a major effect. The intensity of the effect is controlled by the *opacity* of the color during the coloring process. Thus, low (resp. high) opacity results in a low (resp. high) intensity effect (see Fig. 3). To apply the intensity effect, we map the minimum and maximum values to the corresponding parameters for each primitive. For our library, we use six primitives to create nine different structures that showcase how one can create varied designs. The primitives are collected from existing literature, reproduced, and adjusted to fit our fabrication setup. Table 1 shows the available primitives and their parameters, with the corresponding references from the literature. Figure 4 illustrates how the different parameters of each primitive affect the resulting toolpath.

Texture structures. A structure is a list of pairs, each consisting of a segment and a primitive. This means that each segment is assigned to a specific primitive type. In other words, the first segment will be of the type of the first primitive, the second will receive the second primitive, and so on. There are two main guidelines for composing a structure. First, the overall aesthetic is created by the *combination* of primitives, therefore their order is significant. Second, since the segments are printed one on top of the other, we must consider their interactions. For example, the structure Wiggly: [None, Wiggle, None, Wiggle] is designed as such, since the Wiggle primitive requires more space in the z-axis. Accordingly, we allow space above and below it, by using the None primitive (See Fig. 5).

4.2 User Interface

Our interface includes three essential components that facilitate a creative workflow, enabling users to plan and refine their design ideas: a texture library, a painting brush and a visualizer view mode. In the suggested workflow, the user opens a pre-existing file and starts coloring it with our tool. Once done, they save the Gcode file and send it to their printer.

4.2.1 The Library. The library is a collection of nine texture structures tested and calibrated for our fabrication setup; therefore, they can be used out of the box. Each texture is represented using a name, an icon, an image, and a color. The icon visualizes the order of primitives in the structure, as seen in Figure 5, and the image is a closeup of a sample of the printed artifact. When choosing the textures, we took into consideration their robustness, focusing on textures that are printable on different model geometries.

4.2.2 Color as an Interface. Our interface contains a painting *Brush* for applying the textures to the model. We borrow the term *Brush* from graphic software such as Adobe Photoshop, where the cursor location assigns colors or other actions to matching pixels. Similarly, we assign color to the mesh faces. Working with color painting has two main advantages: first, coloring is an intuitive interaction that comes easily to most users since it is widespread in graphic design software. The second advantage is the visual feedback of the users' actions, which allows adjustments such as erasing or overriding. Each unit structure in the interface is paired with a color. In the interface, the user picks a texture from the library. Then, the user

Name	Description	Printing parameters	Intensity effect	Literature
None	Delete the segment and leave a space by skipping it	No extrusion	Not relevant	[50]
Thick	Generate a thick segment	Extrusion: over-extrusion, Speed: Regular	Higher extrusion amount	[57]
Wiggle	Lift the segment in the z-axis, and extrude material in midair, letting it drool and wiggle	Extrusion: over-extrusion, Speed: slow	Higher extrusion amount	[30, 58]
Dot	Subdivide segment into several sections, generating a blob at each point	Extrusion: over-extrusion, Speed: very low speed	Diameter length	[26]
Hair	Generate thin hair-like structures by pulling the material away from the model	Over extrusion for the hair base and under extrusion for the hair stretch	Length of hair	[16]
Arc	Generate an arc in midair by moving away from the model and back towards it	Extrusion: regular, Speed: regular	Length of arc	[6, 8]

Table 1: Texture primitive details: geometric definition, printing parameters, intensity parameter, and related literature.

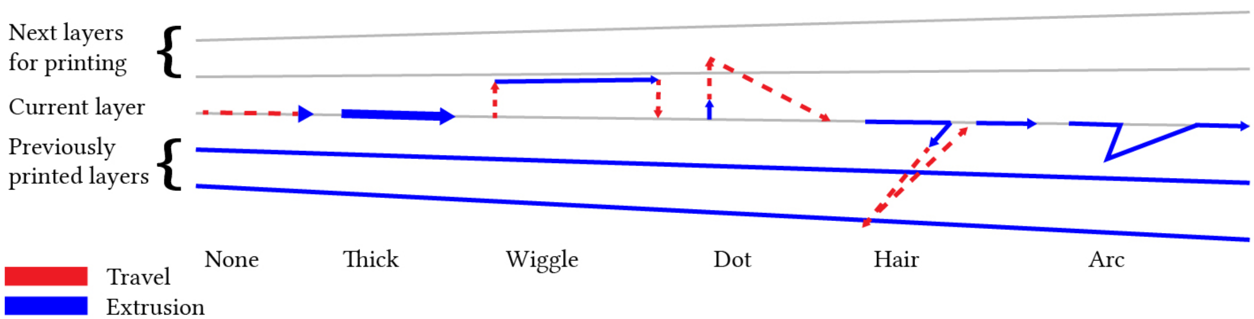


Figure 4: An illustration of the primitives' toolpath changes on the current layer being printed, from left to right. Blue lines represent extruded material, red lines represent travel lines. See Sec. 4.1.

clicks on the "Paint" icon to start painting the 3D models with the color associated with the chosen texture (Fig 6).

In addition to color, the brush size and opacity are adjustable. The brush size is tuned with a slider, which controls the radius within which faces are colored. The opacity slider tunes the intensity of the texture. If the user selects a low opacity and repeatedly colors a face, the opacity accumulates and becomes stronger. After painting, the user confirms their changes by pressing *Apply*. The painting mechanism is based on RhinoCommand MeshPaint [12], which supports our requirement for the adjustments of color, coverage radius, and opacity.

Color coding is a known method of organizing and managing data in textile design programs [51] and in other fields, such as gradual-material printing, where colors are used to represent the flexibility or rigidity of the material. Another example is the DXF file format for laser cutting, where colors are used to mark different actions like cut-out lines and engravings. Furthermore, color opacity is often referred to as a mapping between a color and a spectrum of properties [46], therefore we find it is suitable for representing the intensity of the texture. For example, for the arc primitive (Fig. 3 (a)), the pink color opacity (left) is mapped to the arc length (right). Similarly, for the thick primitive (b), the more opaque the color, the more material is extruded.

4.2.3 Visualization. Introducing the visualization mode is an important step in making the printing outcome more understandable. The need for a tool showing expected results in fabrication processes is evident in both the interviews conducted by Yildirim et al.

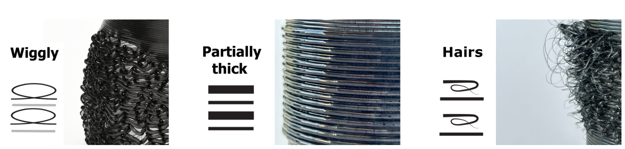


Figure 5: An example of three texture structures: a close-up of the printed structure (right), their name, and icon symbol representation in our library (left).

[67] and in Kim et al. [28]. The challenge of harnessing 3D printing expressive potential by grasping digital-physical correlation is discussed in SketchPath [14]. Therefore, we chose to provide visual feedback through a visualization mode where the user can inspect the expected results.

We chose to implement the visualization as a symbolic representation (see Sec 4.3). Namely, the objective of the visualization is not to provide a physical simulation of the material but rather to give a general impression of the printed outcome. This is shown in Figure 7 where each pair of physical and visualized textures is presented side by side to show the similarities and differences. This implementation suits our system since Finite Element simulation methods for FDM 3D printing are too computationally expensive for an interactive tool [39].

Current Gcode visualizers such as ncviewer[36], Prusa, and Zupfe, display the toolpath as a thin line or pipe showing the printer head's movement, with color-coding to indicate different types of actions, like extrusion or travel (see Fig. 8 a) taken from ncviewer. However, they do not provide insight into the material behavior after being extruded, e.g., the effect of gravity. Grasping how lines translate into material behavior is not straightforward and remains challenging, even for experienced users, due to the complex interplay of various factors such as printing parameters, material properties, and physical forces. To master those behaviors, professionals often iterate the process multiple times [22]. Due to this lack of clear correlation, we found that including the toolpath itself can overburden the presentation, and thus we only show the symbolic representation.

The wiggly texture highlights the challenge in visualization. The Wiggle primitive manipulates the segment by raising it along the z-axis, reducing print speed, and increasing the extrusion amount. A Gcode visualizer would display the segments "floating" above the original curve (see Fig. 8 (a)). In contrast, the expected result is a fuzzy thread following an entangled path, as depicted in our visualization (b). The printed model (c) aligns more closely with our visualization.

Due to the computation time, the visualization is not done continuously but on demand: the user can switch between the coloring

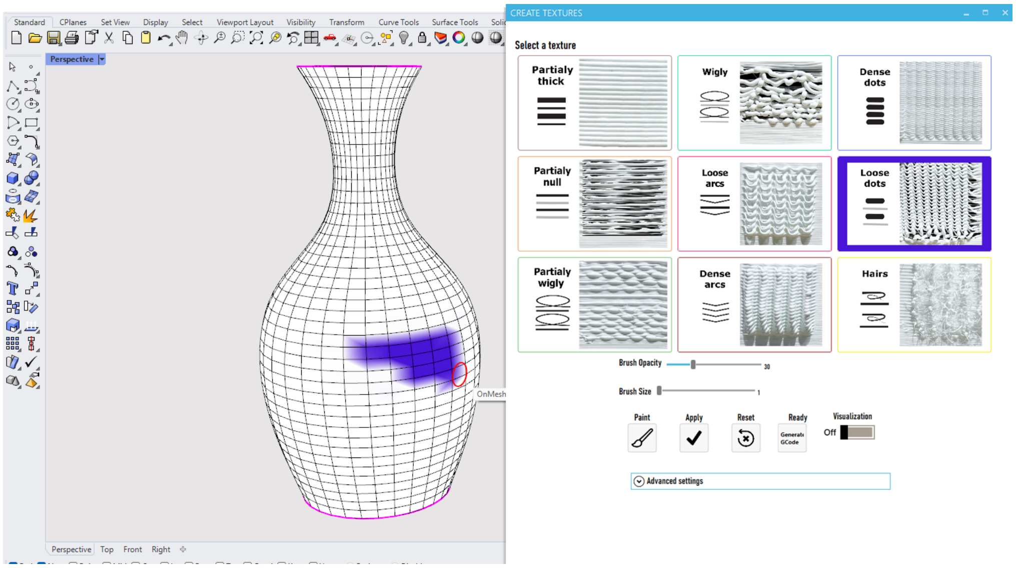


Figure 6: The user interface: The 3D view of the model in the Rhino environment (left). The texture structures library with nine designs, the sliders for brush opacity and size, the action buttons, and the visualization mode toggle (right).

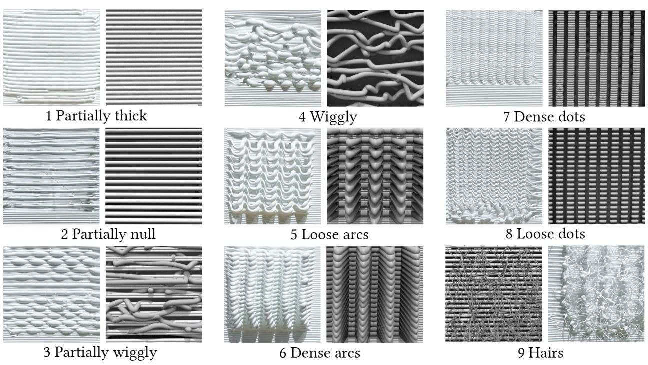


Figure 7: Texture structures and their visualization: nine corresponding pairs of the textures printed with white PLA (left) and the visualization representation in the software (right)

mode and the visualization mode by ticking the "show visualization" checkbox (see Fig. 6). The visualization is the heaviest part of the algorithm, and a complex texture can lead to longer rendering times as discussed in the visualization implementation 4.3.

4.3 Implementation Details

Our pipeline starts with a 3D model represented using a polygonal mesh, we do not re-mesh and are agnostic to whether it contains triangles or quads, since the segmentation is based on the isocurves. After the user adds the textured areas using the brush, we generate the toolpath as follows (see Fig. 2).

Extract Textured Regions. We generate a separate mesh for each textured region by grouping the colored faces (Fig. 2 (a)). Each such region initially contains the iso-z curves that intersect the region (Fig. 2 (b)), which are obtained using the standard planar iso-z slicing.

Partition into Units. We discretize each region into a grid of units, where each unit contains k different z values, and each iso-z curve in the region is divided into the same number of units (Fig. 2 (c)). The number of units for each iso-z curve is obtained from the required segment length c, which depends on the physical parameters. In all our examples, we take $k = 4$ and $c = 5$mm. Finally, we remove extraneous curves (e.g., too short, do not form a full unit, etc.). If we identify a "split" in the region, namely that the number of iso-z curves for the same z value increases or decreases, we separate the region into sub-regions and partition each into units separately.

Apply Primitives. For each unit, we modify each segment according to its corresponding primitives. For example, for the pink texture from Figure 6, we apply the "Arcs" primitive for all 4 segments of the structure (Figure 2(d)). The intensity parameter of the segment is derived from the opacity of the closest face.

Generate Toolpath and Gcode. First, segments that do not belong to any unit are marked as "Regular", meaning they remain unchanged. Then all the curves are reordered to have consecutive z values, and to reduce travel movement where possible. Finally, we apply the geometric and printing parameters modification to all the segments and generate the corresponding Gcode.

Infill. We support printing with infill by applying a raster pattern to cover the internal area of each layer. Infill is treated as an additional primitive within the system, with lines designated as 'infill' having specific parameters similar to other primitives. The use of infill can lead to two potential types of collisions: first, a collision may occur when transitioning from the end of an infill line to the beginning of the skin on the same layer; second, the

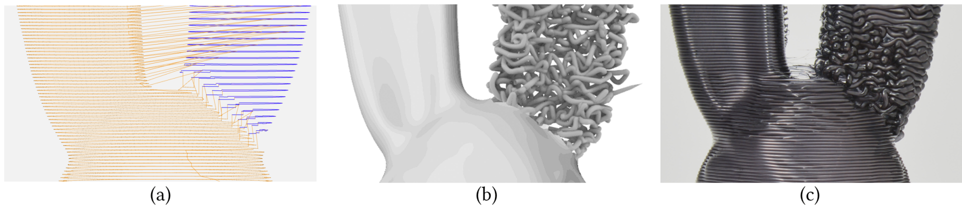


Figure 8: A comparison between a toolpath visualizer and our visualization method for the wiggly-textured ear of the bunny. (a) The toolpath of the produced Gcode file. (b) Our visualization. (c) The output of the printing process. Note that our visualization is more qualitatively similar to the printed model.

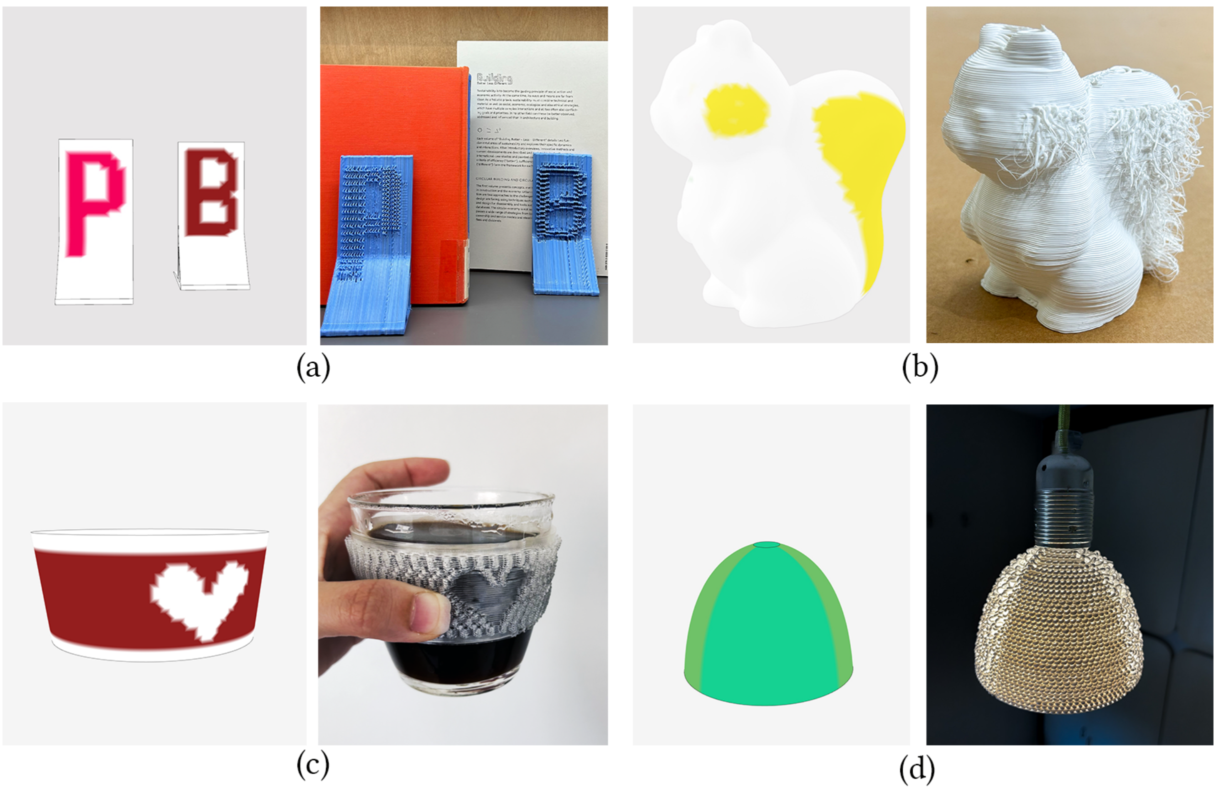


Figure 9: A collage showing four different designed objects. (a) Book stands with the letters P and B. (b) Squirrel with hairy tail and cheeks. (c) Hot cup of coffee with TPU sleeve decorated with a heart. (d) A lampshade with a light effect.

infill may overlap with the skin. To address the first collision type, we incorporate a safety buffer distance along the z-axis. We leave the second type for future work as it requires improved material prediction. An example of a model printed with infill is shown in Figure 11.

Visualization implementation. To create the visualization, we chose to use a symbolic representation. Hence, for each texture primitive there is a corresponding *primitive symbol*, designed according to the visual appearance of the texture. The primitive symbol is generated by a geometric transformation defined for each type of primitive. For example, for the 'thick' primitive, no change is applied to the segments. A more complex symbol is made for 'wiggly': we sample points along the original segment, and for each point we add a random distortion in a predefined radius around the original point. Then we interpolate the points to create the deformed curve. Later, we take all the symbols and create a cylinder along the curve with a radius that corresponds to the extrusion amount of the primitive. For example, a thicker radius for the 'thick' primitive. Additionally, the intensity changes the visual appearance according to its predefined effect (see Table 1). For example, the intensity increases the radius of distortion of the 'wiggly' symbol.

When the visualization mode is turned on, we render the cylinders and display them in addition to the areas of the model that haven't been modified. We use the Rhino 3D environment and engine for rendering, and we do not use an offline rendering option, therefore the rendering is done only on demand. The rendering time varies between a few seconds to around a minute. For example, the vase designed by P10 (Figure 12) took 55 seconds, the lamp shade (Figure 9 (d)) took 37 seconds, and the squirrel (Figure 9 (b)) 22 seconds.

5 DESIGN EXAMPLES

We tested our tool on downloaded models from the online models library Thingiverse [31]. This test serves two purposes: first, checking the performance of the tool on different morphologies. Second, exploring the customization possibilities with existing models.

We provide an example of a stamping roller used for embossing a pattern on soft materials such as clay, Play-dough, paper, and others. The online files for the rollers include a handle and a separate model for each texture. For example, in the collection made by the username '3D-mon', there are ten variations for rollers with texture, which fit on the same handle. Each roller has a specific texture and is not adjustable. Other roller files have a customizer that can add a text pattern to the roller. However, with our tool, we show options for local area patterns printed to fit the original file [1], easily creating variations with a pattern painted by the user (Figure 10).

Home printers are often used for home goods and decorations such as lamps, hence we printed a lamp shade to present the aesthetic value that a texture can add to create an enticing lighting effect using a simple painting pattern (See Fig. 9 (d)). Another model is a blank book stand [25] that we customized by adding letters with different coloring and textures to fit for alphabetic ordering (Fig. 9 (a)). A model of a squirrel [63] was painted yellow on its tail and cheeks to add a hairy surface (See Fig.9 (b)).

Additionally, to demonstrate a workflow where the user has more experience in 3D printing and machine code understanding, we wanted to test our setup with a different material. We tested our software for printing TPU, which is a flexible material. We made the necessary adjustments to accommodate the material requirements, including the material printing temperature and modifications to the speed and extrusion. Those changes are available through the interface advanced settings as shown in Figure 6. For example, changing the material to TPU requires a higher nozzle temperature, a slower speed, and a higher flow rate. We currently allow those changes through the 'advanced setting' menu, which has temperature (head and bed) input fields and sliders for the speed and flow rate. The slider values are relative to the settings value of PLA, meaning that 1.0 is the same speed, and 1.1 is 110% of the speed.

As an example of what can be printed with flexible textures, we substituted a standard hot cup silicone sleeve with a custom-made sleeve printed from high-performance TPU. This sleeve features an overall texture and a heart-shaped negative space (see Fig. 9 (c)). We also printed a bike handlebar grip using the same material, incorporating infill to demonstrate how the texture can fulfill practical needs, such as enhancing grip friction (See Fig. 11).

These examples showcase various functional applications for adding textures onto existing 3D models using our tool. This workflow allows the customization and personalization of existing models, which is a desired feature among FDM printing users as discussed by Alcock et al. [3].

6 USER STUDY

To evaluate our tool, we conducted a user study providing quantitative data on how the users perceived the tool design, highlighting strengths and areas for improvement.

6.1 Methodology

To assess our tool's impact on users' creative experience, we conducted a user study based on a hands-on workshop and a questionnaire. In particular, we ask *how does the tool impact the users' creative experience?* To answer this question, we focused on the users' expressive experience, their level of engagement, and satisfaction with the process and the results. We hypothesize that a tool that lowers the knowledge threshold and technical barriers for interacting with printed textures would provide users with a more fulfilling creative experience.

To find participants, we sent an open call on an online message board for students at our university. We required a minimal level of experience with 3D environments. This requirement is due to a previous study, where we noticed that users without such an experience cannot navigate inside a 3D environment. For instance, they find it difficult to rotate the model, which is required for painting with our brush. We invited 10 participants to a workshop in the lab, working with the tool and rating their experiences. The participants signed a disclosure letter, and the process was approved by the university's ethics committee.

In the workshop, each participant went through two sessions; the first was a moderated walk-through of the tool, and the second was an independent creative session. Each participant spent around an hour in the workshop. The walk-through objective is to introduce the participants to the interface, demonstrating how to apply the textures, control the brush, and turn on the visualization. At the end of the walk-through, participants had time to ask questions and clarify any misunderstandings. For the creative session, the participants were asked to design textures on a vase model we selected (See Fig. 12). The users did not have access to the advanced menu of the interface, since all of their prints were done using PLA material, without having to change material settings. We recorded the interaction on the display during each session for further insights into the process.

To compose the questionnaire, we followed the creativity support index (CSI) by Cherry et al. [7] and Shneiderman's principles of creativity support tools [47]. The CSI defines six measurement criteria to assess the creative support that a tool provides: Exploration, Expressiveness, Immersion, Enjoyment, Results Worth Effort, and Collaboration. Researchers commonly use it to measure how well a tool assists a user who is engaged in a creative task (see e.g., the evaluation of the painting tool developed by Hatley et al. [20, 56] and the hybrid use of 3D pen and printer by Takahashi et al. [56]). We removed CSI questions regarding the collaborative performance of the tool since they are irrelevant in our case.

Our questionnaire contains 15 five-level Likert scale statements and two open-ended questions. Answering the questionnaire is as follows: before the session starts, the user answers the questions about their level of expertise in 3D modeling, 3D printing, and coding. After the session is over, the participant answers the remaining questions and the two open-ended questions; Q16: "What

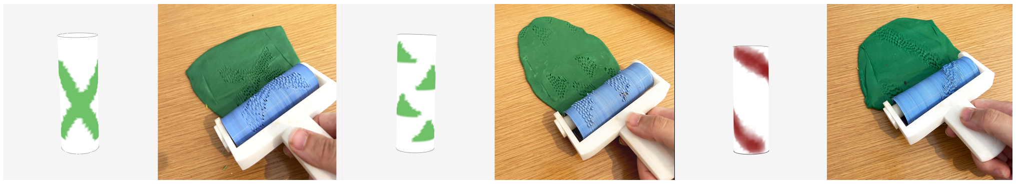


Figure 10: Three examples of stamping roller patterns: The model painted with the tool (left) and the stamped play dough with the resulting pattern (right).

(a) (b) (c) (d)

Figure 11: (a) The painted model (b) The printed model made of TPU (c) zoom in on the texture (d) A flexible bike handle with additional texture for better grip.

were the challenges working with the tool" and Q17: "Anything you like to add about the experience and thoughts:". The questions allow the participants to give their in-depth feedback and go into detail about their subjective perception of the process. Lastly, after we printed the model, we contacted the participants to answer the last two questions regarding their satisfaction with the results. The participants did not watch the printing process, to save their time, therefore we are missing their insights and thoughts on the material's behavior in real-time, for example, did it meet their expectations.

The artifacts produced by the participants in the workshop are distinct, even though they are all based on the same 3D model. By inspecting the coloring patterns the participants created using the brush, it is evident that each participant expressed their design intent differently from the others. Still, we can recognize a few different strategies. P5 and P8 created a full circle horizontal painting around the vase, while P1 followed strictly vertical lines. P6 and P7 combined areas that are mostly either horizontal or vertical. P3 and P10 kept a mostly symmetric pattern around the vase. Other participants, such as P2, P4, and P9, colored in a more free-styled manner.

To learn more about our participants' backgrounds, we asked three questions about their level of experience with 3D printing, 3D modeling, and coding. The options are on a five-step Likert scale, where 1 is beginner and 5 is expert. While most participants had above-medium experience in modeling, most had beginner-level experience with coding, as shown in Figure 13.

6.2 Results

We analyzed the questionnaire's results according to the five criteria suggested in the CSI guideline. Each criterion is represented by two questions, and we average the score of both questions to asses the overall success. The score of each criterion is as follows (see Table 2): Enjoyment (Q4,5): 4.15, Exploration (Q6,7) 3.9, Expressiveness (Q10,11) 3.9, Immersion (Q12,13) 3.6, and Results worth effort (Q14,15) 4.2. Additionally, we added a criterion not included in the CSI guidelines to assess the tool's readability and clarity (Q8,9) 4.35. The answers were on a scale of 1 to 5 where 1 stands for strongly disagree and 5 is strongly agree. The answers to the open questions were transcribed as well (Q16,17). The full results of the questionnaire are listed in Appendix A. Below is our analysis of each criterion.

Enjoyment. For the Enjoyment criterion, we received a score of 4.2 with a standard deviation of 1.2 (Q4) and 0.8 (Q5). The feedback given by P1 and P7 for question Q17 "Anything you like to add about the experience and thoughts" matches the overall high score, stating that *"It's nice and super fun"* and *"I had fun. Very nice."*.

Exploration. For the Exploration criterion, we received a score of 3.9 with a standard deviation of 0.8 (Q6) and 0.9 (Q7). The exploratory experience is supported by the video captures, which show that a few users chose to restart their design. Two of the participants (P4, P8) even asked to create another version and print a few vases instead of just one, and both indicated they were very satisfied with their results (5 / 5 in Q14,15).

Clarity. For the Clarity criterion, we received a score of 4.35 with a standard deviation of 1.4 (Q8) and 0.3 (Q9), which confirms that the majority of participants found the texture selection and application method clear (see results in Fig. 2, Q9). There is a great difference between the results for Q8 and Q9: while nine out of ten users indicated that the texture selection and application method was very clear, half the users stated that the visualization helped them very much, but three stated it only helped a little (2 / 5). We attribute the difference to the visual information presented and its familiarity. When coloring the model, the user views the painted

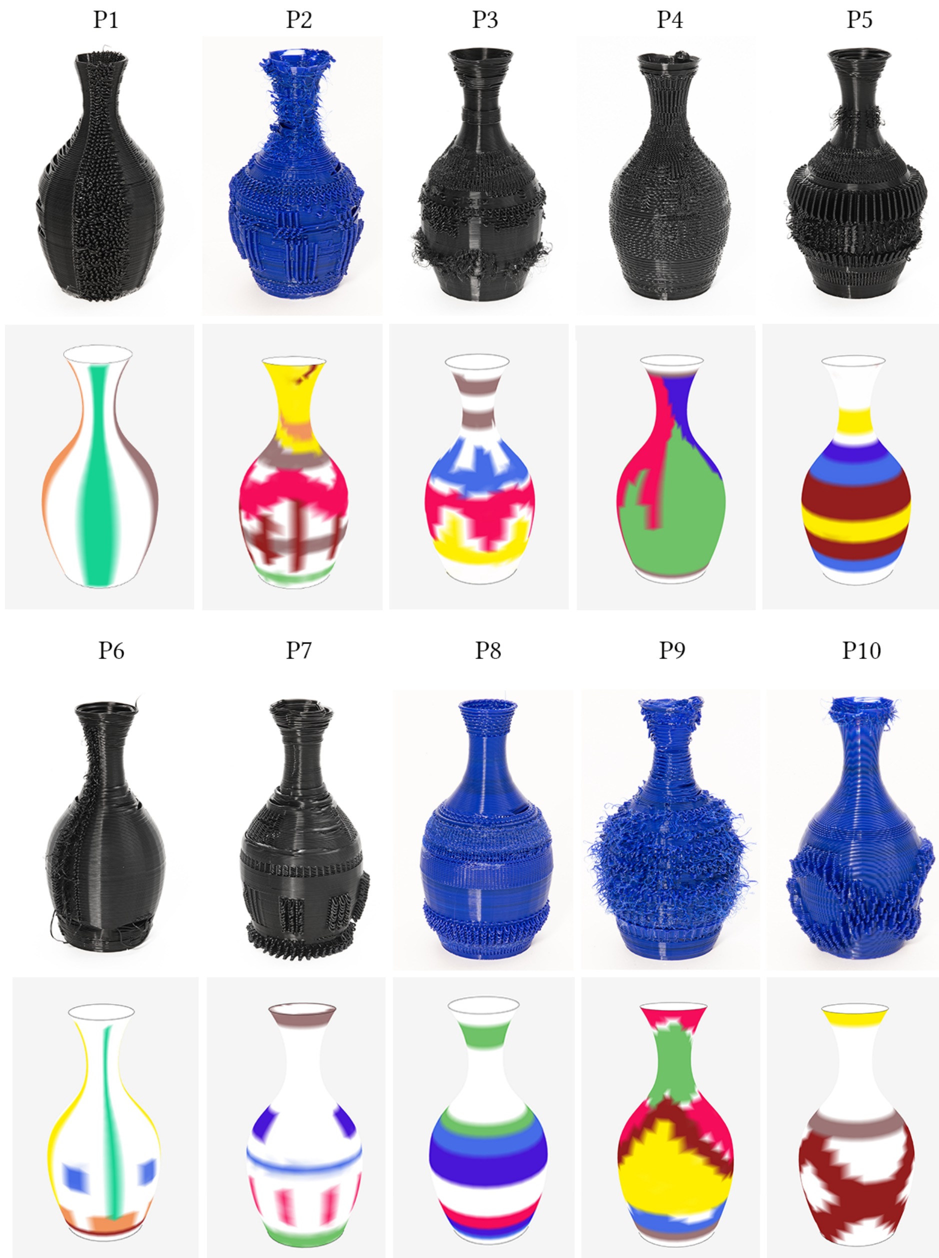


Figure 12: Nine vases were designed in the user study by the participants (P1-P10); for each vase, we show the printed model (top) and the colored model (bottom). Note the varied designs that the participants achieved with our tool.

model. However, when transitioning to the visualization mode, the user views a 3D geometry that represents the expected output. This is the first time the users have seen this form of visualization, and it might require more time to gain confidence in using it.

In the open questions, a few users indicated that they found the meaning of color opacity vague. Participants explained that they understand that opacity is translated into intensity but do not fully understand *how* the intensity is interpreted into texture. Due to the differences between the textures, some correlations were clearer than others. On one hand, for arcs, it was clear that higher intensity leads to longer arcs. On the other hand, when it comes to "none" or "dots", it is not as intuitive how to map the values of one onto the other.

Expressiveness. For the Expressiveness criterion, we received a score of 3.9 with a standard deviation of 0.9 (Q10) and 0.9 (Q11), Yet, we find a few positive indications that the process allowed an expressive experience. First, in the open questions users suggested

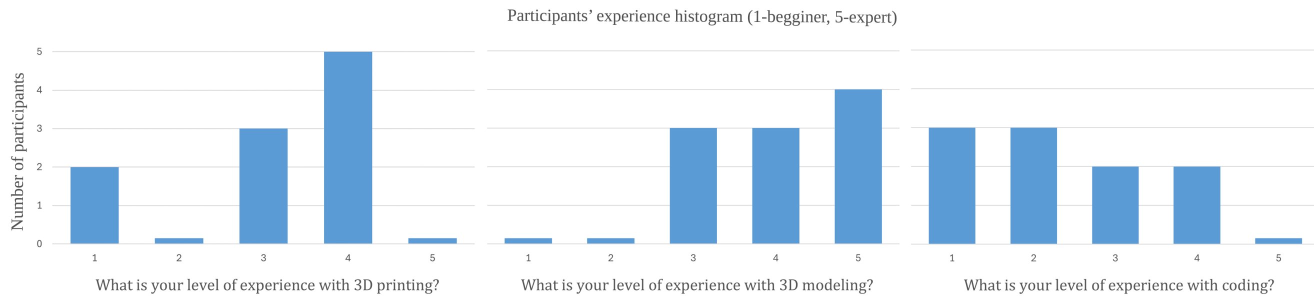


Figure 13: Answers to Questions 1-3: The number of participants who self-identified at each level of experience from beginner (1) to expert (5).

Criteria	Question	Average	Criteria average
Enjoyment	Q4: I would be happy to use this system or tool on a regular basis	4 (±1.2)	4.15
	Q5: I enjoyed using the system or tool	4.3 (±0.8)	
Exploration	Q6: It was easy for me to explore many different ideas,options,designs,or outcomes, using this system or tool	3.7 (±0.8)	3.9
	Q7: The system or tool was helpful in allowing me to track different ideas, outcomes or posibillities	4.1 (±0.9)	
Clarity (ours)	Q8: The visualization helped me understand the structures	3.9 (±1.4)	4.35
	Q9: The textures selection and application method were clear to me	4.9 (±0.3)	
Expressiveness	Q10: I was able to be very creative while doing the activity inside this system or tool	3.9 (±0.9)	3.9
	Q11: The system or tool allowed me to be very expressive	3.9 (±0.9)	
Immersion	Q12: My attention was fully tuned to the activity, and I forgot about the system or tool that I was using	3.9 (±1.1)	3.6
	Q13: I became so absorbed in the activity that I forgot about the system or tool that I was using	3.4 (±1.2)	
Results worth effort	Q14: I was satisfied with what I got of the system or tool	4.0 (±1.1)	4.2
	Q15: What I was able to produce was worth the effort I had to exert to produce it	4.4 (±0.8)	

Table 2: Results of the questionnaire: for each criterion, relevant questions scores are averaged and calculated for their standard deviation. We also show the average score for each question and for each criterion.

improvements that would allow them to execute their designs. P7 wrote in the challenges that *"There is no option for automatic symmetry or square brush/area selection"*, P10 wanted more control over the texture design, giving the example of the spaces between the dots in the dots texture. This means users found some of the features missing in our implementation to be a barrier to their self-expression. Secondly, we identify different coloring design patterns emerge between the participants, which leads us to believe the tool does not subject users to a single design style.

Immersion. For the Immersion criterion, we received a score of 3.6 with a standard deviation of 1.1 (Q12) and 1.2 (Q13). This is the lowest scoring among the different criteria, and we attribute it to several factors. First, the workshop was short, taking around one hour for each participant which might not be sufficient for immersing in a process. Second, the somewhat slow runtime also damaged the immersion. As described in the answer to the question about challenges, P9 mentioned the pauses, *"Pause between each texture time to load (both brush and visualization)."*

Results worth effort. For the Results worth effort criterion, we received a score of 4.2 with a standard deviation of 1.1 (Q14) and 0.8 (Q15). Even though participants complained about the time the software took to render their design, they still found the effort worth the wait. Some users could point out the differences between their design and the output and find bugs with the software. For example, P10 (See Fig. 12) noticed some patches that were missing in the pattern, stating in Q16 that "the coloring identification wasn't accurate", which might explain their medium-level (3/5) satisfaction with the result. Furthermore, participants with a strong modeling background of 4 and above, who we assume have a better idea of how to execute their design through modeling, indicated that working with the tool was worth the effort (Q15) and scored on average. 4.3 out of 5. Finally, despite the visualization's limitations, when the users received their designs after several days, they were able to recognize their work. This indicates that they have developed a mental image of their expected results, even though it was the first time that they saw the printed model.

7 DISCUSSION

Critics of design tools that automate and simplify production processes often argue they might undermine user agency and control, reducing users to passive operators. On the contrary, we argue for a balance between more agency to the user in design decisions, while automating non-creative aspects of the process such as generating a proper gcode file. Additionally, the use of CAD and slicer software in the design process can obstruct design intentions as discussed in Goudswaard et al. [18], therefore requiring a negotiation between designer intent and machine automated processes. We build upon previous arguments supporting the development of CAM-focused tools, to provide design freedom and exploration experiences. This is supported by the results of our user study, showing positive feedback for enjoyment, exploration, and expressiveness with the use of ConTextural.

When looking at the participants' backgrounds, finding a meaningful correlation between one of the knowledge fields and their scoring is hard. It is slightly noticeable that participants with beginner (1/5) experience in coding (P1, P4, P8), are the group that on average gave the highest score. Additionally, participants with a beginner level (1/5) of experience in 3D printing (P3, P7) also scored higher than their peers. Interestingly, participants with an expert level of experience in 3D modeling (P2, P6, P10) had a lower score for Enjoyment, and on average, a lower score as a group compared to their peers. This can potentially be the result of the desired level of abstraction different skilled leveled groups desire. Relevant literature, such as the work of Yildirim et al. [67], supports the desire of fabrication professionals to customize their tools, both enjoying automation options while keeping control when needed.

While existing research includes drawing tools to design layer paths, like SketchPath [14], we are unfamiliar with painting-based methods of applying changes to existing models. Therefore, we added the "clarity" criteria, to test whether our assumption of painting being a clear interaction method is valid. All users indicated that the texture application method was clear; therefore, we suggest that this method should be further developed in the pursuit of a simple and intuitive interaction method. Additionally, we find the diverse patterns of coloring done by users an indication that self-expression is possible with the coloring brush and think the brush acts upon mesh faces similarly to how it can manipulate pixels in 2D mediums in more use cases.

By examining users' designs and design examples, we identify textures and forms that cannot be created through traditional CAD modeling. By enabling design possibilities beyond the limitations of material predictions in digital modeling [60], we believe the tool contributes to broadening the 3D printing design space. However, we also acknowledge the benefits of modeling for self-expression, along with the growing popularity of file-sharing platforms and remixing behaviors among 3D printer users. As a result, this work proposes a hybrid approach that combines elements of both CAD and CAM mindsets.

7.1 Limitations and Future Work

ConTextural is a prototype developed to test a proposed workflow for adding textures onto 3D models, mainly for 3D printing users without coding experience. To allow researchers and experienced coders to extend the existing library of available texture primitives and texture structures, we plan to add a texture importer and texture editor, thereby exposing internal features of the software that were not exposed through the prototype. These editors will be accompanied by detailed protocols and documentation for each parameter, which would allow simple integration of new textures. Additionally, we intend to integrate Contextural for other of extrusion-based systems such as clay 3D printing. We plan to collaborate with ceramicists, who have a strong understanding of material aesthetics and design, but may lack technical coding skills. Such collaboration would allow us to test how the tool can serve different creative needs and fabrication setup integrations, which will potentially lead to the definition of new textures, and allow us to further extend the existing library.

ConTextural has a few limitations that can be addressed in future work. First, our definition of primitive geometric manipulation is restricted to the XY plane, which limits the applicability to certain geometries such as the horizontal planes on the model's top. This can potentially be addressed by changing the primitive definition according to the model's face orientation along the XZ and YZ axes. Second, since we divide each isocurve into an equal number of segments, there could be a variance in their length, which can lead to an inconsistent texture appearance. Lastly, the rendering time of the visualization method we chose slows down the design iterations. To improve run time in the future, we wish to test different visualization methods with different graphical representations to better understand the trade-off between runtime and the visualization's impact on the design process and the user's experience with the tool.

Additionally, our method has physical constraints that derive from the fabrication process. First, the coverage area has a minimal size, for example, at least one structure unit. The physical size of the structure is also bounded: we chose to limit its width to at least 2mm. Those dimensional limitations are related to the extrusion line thickness and to the nozzle diameter. Furthermore, as explained in the discussion about infill in Sec. 4.3, there are potential collisions that are hard to predict and avoid, for example, the wiggly texture can bend in unpredictable ways, making it overlap with the infill.

Moreover, ConTextural can be generalized in a few ways. First, we wish to expand the range of brushes supported in the system by drawing inspiration from other unit-based, brush-based software, such as Adobe Photoshop. Additionally, we want to address user-suggested features related to selection options and automatic completions, such as applying symmetry or having a shape/area selection option. We believe focusing on the brush capabilities will significantly enhance the tool's functionality and expose new texture application options.

Finally, we see the potential of applying the mesh coloring-based workflow for other physical properties of models using toolpath manipulation. For example, this interface may be used for locally controlling the softness and stiffness of models printed with flexible materials. By adjusting the extrusion amount or other printing parameters, the models' softness may be locally tuned. This approach will provide an abstraction level to a vast space of properties through a simple interface. Some of these properties are not accessible with the current CAD and slicer workflow.

8 CONCLUSIONS

This paper introduces an infrastructure for conceptualizing textures as discrete units in a modular and extensible framework. We show that despite their inherent unpredictability, textures derived from toolpath manipulation provide a substantial aesthetic value and a broad design space. We develop a design tool to allow access to that design space, through a simple coloring interface, abstracting the complexity of the toolpath. We show that this workflow actively engages users in a creative and exploratory process, enabling them to venture into design domains currently inaccessible to them due to a lack of knowledge and experience in the field. We believe computational design tools, such as the one presented here, have the potential to lower the bar for engaging with digital fabrication tools, thereby empowering users to create expressive and personalized artifacts.

ACKNOWLEDGMENTS

Daphna Kaplan and Mirela Ben Chen acknowledge the support of the Israel Science Foundation (grant No. 1073/21).

REFERENCES

[1] 3D-mon. 2024. *thing:6690924*. Thingiverse. https://www.thingiverse.com/thing:6690924
[2] Lea Albaugh, James McCann, Lining Yao, and Scott Hudson. 2021. Enabling personal computational handweaving with a low-cost jacquard loom. In *Conference on Human Factors in Computing Systems - Proceedings*. Association for Computing Machinery, New York, NY, USA, 1–10. https://doi.org/10.1145/3411764.3445750
[3] Celena Alcock, Nathaniel Hudson, and Parmit K. Chilana. 2016. Barriers to using, customizing, and printing 3D designs on thingiverse. In *Proceedings of the International ACM SIGGROUP Conference on Supporting Group Work*, Vol. 13-16-November-2016. Association for Computing Machinery, New York, NY, USA, 195–199. https://doi.org/10.1145/2957276.2957301
[4] Davide Jose Nogueira Amorim, Troy Nachtigall, and Miguel Bruns Alonso. 2019. Exploring mechanical meta-material structures through personalised shoe sole design. In *Proceedings: SCF 2019 - ACM Symposium on Computational Fabrication*. Association for Computing Machinery, Inc, New York, NY, USA, 1–8. https://doi.org/10.1145/3328939.3329001
[5] M. F. Ashby and Kara Johnson. 2014. *Materials and design : the art and science of material selection in product design Mike Ashby and Kara Johnson* (3rd ed. ed.). Elsevier Science ; Butterworth-Heinemann, Oxford.
[6] Samuelle Bourgault, Pilar Wiley, Avi Farber, and Jennifer Jacobs. 2023. CoilCAM: Enabling Parametric Design for Clay 3D Printing Through an Action-Oriented Toolpath Programming System. In *Conference on Human Factors in Computing Systems - Proceedings*. Association for Computing Machinery, New York, NY, USA, 1–16. https://doi.org/10.1145/3544548.3580745
[7] Erin Cherry and Celine Latulipe. 2014. Quantifying the creativity support of digital tools through the creativity support index. *ACM Transactions on Computer-Human Interaction (TOCHI)* 21, 4 (2014), 1–25.
[8] James Clarke-Hicks, Isabel Ochoa, and David Correa. 2023. Harnessing plastic deformation in porous 3D printed ceramic light screens. *Architecture, Structures and Construction* 3, 2 (6 2023), 193–204. https://doi.org/10.1007/s44150-022-00079-0
[9] Zach Cohen. 2019. *Hold up: machine delay in architectural design*. Springer.
[10] Yuran Ding, Craig Shultz, and Chris Harrison. 2023. Surface I/O: Creating Devices with Functional Surface Geometry for Haptics and User Input. In *Conference on Human Factors in Computing Systems - Proceedings*. Association for Computing Machinery, New York, NY, USA, 1–22. https://doi.org/10.1145/3544548.3581037
[11] Frikk Fossdal, Rogardt Heldal, and Nadya Peek. 2021. Interactive Digital Fabrication Machine Control Directly within a CAD Environment. In *Proceedings - SCF 2021: ACM Symposium on Computational Fabrication*. Association for Computing Machinery, Inc, New York, NY, USA, 1–15. https://doi.org/10.1145/3485114.3485120
[12] Ursula Frick and Thomas Grabner. 2012. *MeshPaint3D*. food4rhino. https://www.food4rhino.com/en/app/meshpaint3d
[13] Camila Friedman-Gerlicz, Deanna Gelosi, Fiona Bell, and Leah Buechley. 2024. WeaveSlicer: Expanding the Range of Printable Geometries in Clay. In *Proceedings of the CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 352, 16 pages. https://doi.org/10.1145/3613904.3642622
[14] Devon Frost, Raina Lee, Eun-Ha Paek, and Jennifer Jacobs. 2024. SketchPath: Using Digital Drawing to Integrate the Gestural Qualities of Craft in CAM-Based Clay 3D Printing. In *Proceedings of the CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, 1–16. https://doi.org/10.1145/3613904.3642684
[15] Gierad Laput, Xiang 'Anthony' Chen, and Chris Harrison. 2015. 3D Printed Hair: Fused Deposition Modeling of Soft Strands, Fibers and Bristles. In *UIST 2015 - Proceedings of the 28th Annual ACM Symposium on User Interface Software and Technology* (2015-11-05). Association for Computing Machinery, Inc, New York, NY, USA, 167–173. https://doi.org/10.1145/2807442.2807484
[16] Gierad Laput, Xiang 'Anthony' Chen, and Chris Harrison. 2015. 3D Printed Hair: Fused Deposition Modeling of Soft Strands, Fibers and Bristles. In *UIST 2015 - Proceedings of the 28th Annual ACM Symposium on User Interface Software and Technology*. Association for Computing Machinery, Inc, New York, NY, USA, 167–173. https://doi.org/10.1145/2642918.2647356
[17] Andrew Gleadall. 2021. FullControl GCode Designer: Open-source software for unconstrained design in additive manufacturing. *Additive Manufacturing* 46 (2021), 102109. https://doi.org/10.1016/j.addma.2021.102109
[18] Maas Goudswaard, Bruna Goveia Da Rocha, and Kristina Andersen. 2024. Entering the 3D printer: negotiations of imprecision in making.. In *Proceedings of the 2024 ACM Designing Interactive Systems Conference* (Copenhagen, Denmark) *(DIS '24)*. Association for Computing Machinery, New York, NY, USA, 1148–1161. https://doi.org/10.1145/3643834.3660758
[19] Benay Gürsoy. 2018. From control to uncertainty in 3D printing with clay. In *Proceedings of the 36th eCAADe Conference: Computing for a better tomorrow*. eCAADe, 21–30.
[20] Jonathan Hatley, Noah Posner, Hyunjoo Oh, and Sang-won Leigh. 2022. Mixed Dimensions: Exploring Novel Material and Extrusion Tool for 2.5 D Shape Painting. In *Proceedings of the Sixteenth International Conference on Tangible, Embedded, and Embodied Interaction*. Association for Computing Machinery, New York, NY, USA, 1–10.
[21] Andrew Heumann. 2016. *Human UI*. private.
[22] Mare Hirsch, Gabrielle Benabdallah, Jennifer Jacobs, and Nadya Peek. 2023. Nothing Like Compilation: How Professional Digital Fabrication Workflows Go Beyond Extruding, Milling, and Machines. *ACM Transactions on Computer-Human Interaction* 31, 1 (2023), 1–45.
[23] Megan Hofmann, Lea Albaugh, Ticha Sethapakadi, Jessica Hodgins, Scott E. Hudson, James McCann, and Jennifer Mankoff. 2019. KnitPicking Textures: Programming and Modifying Complex Knitted Textures for Machine and Hand Knitting. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* (New Orleans, LA, USA) *(UIST '19)*. Association for Computing Machinery, New York, NY, USA, 5–16. https://doi.org/10.1145/3332165.3347886
[24] Alexandra Ion, Robert Kovacs, Oliver S. Schneider, Pedro Lopes, and Patrick Baudisch. 2018. Metamaterial textures. In *Conference on Human Factors in Computing Systems - Proceedings*, Vol. 2018-April. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3173574.3173910
[25] jorge_roma. 2021. *thing:4730791*. Thingiverse. https://www.thingiverse.com/thing:4730791

[26] Daphna Kaplan and Yoav Sterman. 2023. Buildup: FDM Method for Printing Textural Surfaces. In *Proceedings of the 8th ACM Symposium on Computational Fabrication (SCF '23)*. Association for Computing Machinery, New York, NY, USA, 1–2. https://doi.org/10.1145/3623263.3629156

[27] Jeeeun Kim, Mark D. Gross, Clement Zheng, Daniel Ashbrook, Haruki Takahashi, and Tom Yeh. 2018. Compositional 3D printing: Expanding & supporting workflows towards continuous fabrication. In *Proceedings - SCF 2018: ACM Symposium on Computational Fabrication*. Association for Computing Machinery, Inc, New York, NY, USA, 1–10. https://doi.org/10.1145/3213512.3213518

[28] Jeeeun Kim, Haruki Takahashi, Homei Miyashita, Michelle Annett, and Tom Yeh. 2017. Machines as Co-Designers: A Fiction on the Future of Human-Fabrication Machine Interaction. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI EA '17)*. Association for Computing Machinery, New York, NY, USA, 790–805. https://doi.org/10.1145/3027063.3052763

[29] Jingyi Li, Eric Rawn, Jacob Ritchie, Jasper Tran O'Leary, and Sean Follmer. 2023. Beyond the Artifact: Power as a Lens for Creativity Support Tools. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 47, 15 pages. https://doi.org/10.1145/3586183.3606831

[30] Jeffrey I. Lipton and Hod Lipson. 2016. 3D Printing Variable Stiffness Foams Using Viscous Thread Instability. *Scientific Reports* 6 (9 2016), 29996. https://doi.org/10.1038/srep29996

[31] MakerBot Industries. 2024. thingiverse.

[32] Zofia Marciniak, Kongpyung Moon, and Andrea Bianchi. 2024. Decoupling Geometry from Surface Finish by Parameterizing Texture Directly in G-code for Fused Deposition Modeling (FDM) Printing. *Archives of Design Research-Vol* 37, 2 (2024), 7–23.

[33] Mark Wheadon. 2017. Velocity printing.

[34] Leo McElroy and Lingdong Huang. 2023. PotScript: a visual grammar for sculpting with functions. In *Proceedings of the 8th ACM Symposium on Computational Fabrication*. ACM, New York, NY, USA, 1–9. https://doi.org/10.1145/3623263.3623357

[35] Ashish Mohite, Mariia Kochneva, and Toni Kotnik. 2019. Speed of Deposition Vehicle for structural and aesthetic expression in CAM. In *International Conference on Education and Research in Computer Aided Architectural Design in Europe: Architecture in the Age of the 4th Industrial Revolution*. eCAADe, eCAADe, Porto, Portugal, 729–738.

[36] Inc. NC Viewer © 2024 Toolpath Labs. 2024. *NC viewer*. NC Viewer © 2024 Toolpath Labs, Inc. https://ncviewer.com/

[37] Paul O'Dowd, Stephen Hoskins, Peter Walters, and Adrian Geisow. 2015. Modulated Extrusion for Textured 3D Printing. *NIP Digital Fabrication Conference* 31, 1 (2015), 173–173. https://doi.org/10.2352/ISSN.2169-4451.2015.31.1.art00038_1

[38] Joyce E Passananti, Ana CardenasGasca, Jennifer Jacobs, and Tobias HÖLlerer. 2023. SculptAR: Direct Manipulations of Machine Toolpaths in Augmented Reality for 3D Clay Printing. In *Adjunct Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*. Association for Computing Machinery (ACM), New York, NY, USA, 1–3. https://doi.org/10.1145/3586182.3615822

[39] Sumit Paul. 2021. Finite element analysis in fused deposition modeling research: A literature review. *Measurement* 178 (2021), 109320. https://doi.org/10.1016/j.measurement.2021.109320

[40] Pawel Pawlus, Rafal Reizer, and Michal Wieczorowski. 2021. Functional importance of surface texture parameters. *Materials* 14, 18 (2021), 5326.

[41] Tatiana Pereira, Sónia Barroso, and Maria M. Gil. 2021. Food texture design by 3d printing: A review. https://doi.org/10.3390/foods10020320

[42] Franklin Pezutti-Dyer and Leah Buechley. 2022. Extruder-Turtle: A Library for 3D Printing Delicate, Textured, and Flexible Objects. In *ACM International Conference Proceeding Series*. Association for Computing Machinery, New York, NY, USA, 28. https://doi.org/10.1145/3490149.3501312

[43] Pollen AM inc. 2023. PollenAM. https://pollen.am/

[44] Robert McNeel & Associates. 2024. Grasshopper. https://www.grasshopper3d.com/

[45] Ronald Rael. 2020. PotterWare by Emerging Objects.

[46] Karen B Schloss, Connor C Gramazio, Allison T Silverman, Madeline L Parker, and Audrey S Wang. 2018. Mapping color to meaning in colormap data visualizations. *IEEE transactions on visualization and computer graphics* 25, 1 (2018), 810–819.

[47] Ben Shneiderman. 2007. Creativity support tools: accelerating discovery and innovation. *Commun. ACM* 50, 12 (dec 2007), 20–32. https://doi.org/10.1145/1323688.1323689

[48] David J Spencer. 2001. *Knitting technology: a comprehensive handbook and practical guide*. Vol. 16. CRC press, Cambridge, UK.

[49] Yoav Sterman and Eli Almog. 2022. A Computational Design Tool for Gradual Transition of Knit Structures in Seamless Circular Knitting. *CAD Computer Aided Design* 146 (5 2022), 103214. https://doi.org/10.1016/j.cad.2022.103214

[50] Yoav Sterman, Shahar Asor, Yuval Gur, and Ezri Tarazi. 2023. *Bridgeslicer: a creative slicer for 3D printing perforated shells*. Technical Report. https://doi.org/10.52842/conf.caadria.2023.2.551

[51] Stoll. 2024. *M1 Plus*. Stoll. https://software.stoll.com/m1plus/QuickStartGuide/en/index.html

[52] Blair Subbaraman and Nadya Peek. 2022. p5.fab: Direct Control of Digital Fabrication Machines from a Creative Coding Environment. In *DIS 2022 - Proceedings of the 2022 ACM Designing Interactive Systems Conference: Digital Wellbeing*. Association for Computing Machinery, Inc, New York, NY, USA, 1148–1161. https://doi.org/10.1145/3532106.3533496

[53] Lingyun Sun, Jiaji Li, Yu Chen, Yue Yang, Ye Tao, Guanyun Wang, and Lining Yao. 2020. 4DTexture: A shape-changing fabrication method for 3D surfaces with texture. In *Conference on Human Factors in Computing Systems - Proceedings*. Association for Computing Machinery, New York, NY, USA, 1–7. https://doi.org/10.1145/3334480.3383053

[54] Ryo Suzuki, Koji Yatani, Mark D. Gross, and Tom Yeh. 2018. Tabby: Explorable Design for 3D Printing Textures. *arXiv preprint arXiv:1810.13251* (10 2018), 0–4. https://doi.org/10.2312/pg.20181273

[55] Alina Surmacka Szczesniak. 2002. Texture is a sensory property. *Food Quality and Preference* 13, 4 (2002), 215–225. https://doi.org/10.1016/S0950-3293(01)00039-8

[56] Haruki Takahashi and Jeeeun Kim. 2019. 3D Pen + 3D Printer: Exploring the Role of Humans and Fabrication Machines in Creative Making. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3290605.3300525

[57] Haruki Takahashi and Homei Miyashita. 2016. Thickness control technique for printing tactile sheets with fused deposition modeling. In *Adjunct Proceedings of the 29th Annual ACM Symposium on User Interface Software and Technology*. Association for Computing Machinery, New York, NY, USA, 51–53.

[58] Haruki Takahashi and Homei Miyashita. 2017. Expressive fused deposition modeling by controlling extruder height and extrusion amount. In *Conference on Human Factors in Computing Systems - Proceedings*, Vol. 2017-May. Association for Computing Machinery, New York, NY, USA, 5065–5074. https://doi.org/10.1145/3025453.3025933

[59] Mert Toka, Devon Frost, Samuelle Bourgault, Avi Farber, Camila Friedman-Gerlicz, Raina Lee, Eun-ha Paek, Pilar Wiley, and Jennifer Jacobs. 2024. Practice-driven Software Development: A Collaborative Method for Digital Fabrication Systems Research in a Residency Program. In *Proceedings of the 2024 ACM Designing Interactive Systems Conference*. Association for Computing Machinery, New York, NY, USA, 1192–1217.

[60] Iremnur Tokac, Corneel Cannaerts, Herman Bruyninckx, and Andrew Vande Moere. 2019. Material sketching: Towards the digital fabrication of emergent material effects. In *Conference on Human Factors in Computing Systems - Proceedings*. Association for Computing Machinery, New York, NY, USA, 1–6. https://doi.org/10.1145/3290607.3313036

[61] Cesar Torres, Tim Campbell, Neil Kumar, and Eric Paulos. 2015. Hapticprint: Designing feel aesthetics for 3D printing. In *UIST 2015 - Proceedings of the 28th Annual ACM Symposium on User Interface Software and Technology*. Association for Computing Machinery, Inc, New York, NY, USA, 583–591. https://doi.org/10.1145/2807442.2807492

[62] Ultimaker. 2023. *Cura*. Ultimaker.

[63] user MBCook. 2011. *thing:11705*. Thingiverse. https://www.thingiverse.com/thing:11705

[64] Bhupesh Verma, Omid Zarei, Song Zhang, and Johannes Henrich Schleifenbaum. 2022. Development of texture mapping approaches for additively manufacturable surfaces. *Chinese Journal of Mechanical Engineering* 35, 1 (2022), 92.

[65] WASP S.r.l. 2023. 3dwasp.

[66] Xin Yan, Lin Lu, Andrei Sharf, Xing Yu, and Yulu Sun. 2021. Man-made by Computer: On-The-Fly Fine Texture 3D Printing. In *Proceedings - SCF 2021: ACM Symposium on Computational Fabrication*. Association for Computing Machinery, Inc, New York, NY, USA, 1–10. https://doi.org/10.1145/3485114.3485119

[67] Nur Yildirim, James McCann, and John Zimmerman. 2020. Digital Fabrication Tools at Work: Probing Professionals' Current Needs and Desired Futures. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3313831.3376621

[68] Tianhong Catherine Yu and James McCann. 2020. Coupling Programs and Visualization for Machine Knitting. In *Proceedings - SCF 2020: ACM Symposium on Computational Fabrication*. Association for Computing Machinery, Inc, New York, NY, USA, 1–10. https://doi.org/10.1145/3424630.3425410

A APPENDIX

We include a table of the raw results collected through the questionnaire in the user study.

Num	type	Question	p1	p2	p3	p4	p5	p6	p7	p8	p9	p10
Q1	Background	What is your level of experience with 3D printing	3	4	1	4	4	3	1	4	3	4
Q2	Background	What is your level of experience with 3D modeling	3	5	3	5	4	5	4	4	3	5
Q3	Background	What is your level of experience with coding?	1	2	2	1	3	3	4	1	2	4
Q4	Enjoyment	I would be happy to use this system or tool on a regular basis	5	2	5	5	4	4	4	5	4	2
Q5	Enjoyment	I enjoyed using the system or tool	5	3	5	4	4	3	5	5	5	4
Q6	Exploration	It was easy for me to explore many different ideas,options,designs,or outcomes, using this system or tool	5	3	4	4	4	2	4	4	4	3
Q7	Exploration	The system or tool was helpful in allowing me to track different ideas, outcomes or posibillities	5	4	3	5	5	3	5	4	3	4
Q8	Clarity (ours)	The visualization helped me understand the structures	5	4	5	5	3	2	5	2	5	2
Q9	Clarity (ours)	The textures selection and application method were clear to me	5	4	5	5	5	5	5	5	5	5
Q10	Expressiveness	I was able to be very creative while doing the activity inside this system or tool	5	2	4	5	4	4	4	4	4	3
Q11	Expressiveness	The system or tool allowed me to be very expressive	5	2	4	5	4	3	4	4	4	4
Q12	Immersion	My attention was fully tuned to the activity, and I forgot about the system or tool that I was using	3	2	3	5	3	4	5	5	3	5
Q13	Immersion	I became so absorbed in the activity that I forgot about the system or tool that I was using	3	3	3	5	2	3	5	5	3	2
Q14	Results worth effort	I was satisfied with what I got of the system or tool	4	2	3	5	5	4	5	5	4	3
Q15	Results worth effort	What I was able to produce was worth the effort I had to exert to produce it	4	4	5	5	5	3	5	5	5	3

Table 3: All the results collected through the questionnaire in the user study