

3D Printing Slicer for Computational Texture Generation

Zofia Marciniak

Industrial Design KAIST
zofia.marciniak@kaist.ac.kr

Kongpyung Moon

Industrial Design KAIST
jkpmoon@kaist.ac.kr

Andrea Bianchi

Industrial Design KAIST
andrea@kaist.ac.kr

Abstract

This paper introduces a novel workflow of 3D printing various finish patterns of an object without CAD modelling an actual surface texture but computationally generating a g-code for 3D printing with an embedded texture. It allows designers to easily convert a simple 3D surface to a textured surface they desire. In this paper, we describe how our slicer generates texture patterns from a given geometry and 3D print 40 samples of textured surfaces to explore the texture design spectrum of our slicer. We also present three applications to demonstrate the feasibility of our texture slicer.

Keyword

3D-printing, computational design, digital fabrication

1. Introduction

3D Printers, particularly Fused Deposition Modeling (FDM) machines, are widely used by designers, researchers and engineers to rapidly prototype their ideas and products. The typical workflow of 3D printing goes from modelling in CAD and exporting to 3D slicers such as Cura, Simplify3d, or PrusaSlicer. These slicers allow users to customize parameters such as nozzle temperature, printing speed, layer width, and height. However, to 3D print textured surfaces, users have to re-open the 3D model, modify the model's surface geometry, and re-export to the slicer. Designers have to go through this iterative and trivial but necessary process to obtain the textured surface they desire.

Designers use numerous finishing textures, such as smooth, glossy, matt, bumpy, and rough, to render a distinct look of a product. Additionally, the surface of a product can transform its properties to have additional features like anti-skid, hook-and-loop, and even varying taste perception.

In this paper, we propose a custom slicer for Rhino Grasshopper that generates various

textures on a surface without requiring modifications of the actual geometry. This is achieved by injecting the g-code during the slicing process before printing. It allows users to generate different texture features, such as tactile feedback based on a single geometry, without remodeling for every texture pattern.

2. Related Works

Our work closely intersects prior works in 3D printing with custom g-code and fabricating textures on a printed object.

2.1 G-code modification for 3D Printing

Custom g-code modification has been extensively used by HCI researchers and in the digital fabrication domain. Programmable filaments introduce multi-color 3D printing techniques with a single nozzle 3D printer [1]. WirePrint 3D prints directly in a 3D space, not layer-by-layer, to maximize the printing speed for rapid prototyping [2]. Extruder-Turtle introduces a set of 3D printed textures by LOGO inspired turtle coding method [3]. Moreover, O'Leary et al., introduces a scripting interface with visual interpreter for generating custom g-code for 3D printers [4]. In a similar manner, we propose *TextureSlicer*, which generates a printing path that enables surface textures from a given 3D geometry without modifying the geometry itself.

2.2 3D printed objects with texture

Multiple works have explored the importance of surface texture with 3D-printed objects. Cillia 3D prints microns scale hair like structures for passive actuators, mechanical adhesion, and touch sensors [5]. 4Dtexture uses multiple materials to generate various textures on 3D printed objects [6]. On the other hand, [7,8,9] have explored the impact of different surface textures on human perception of taste when drinking beverages. Despite the importance of how texture influences a product's design and features, users have to re-3D model and export it to the slicer repetitively until the final design.

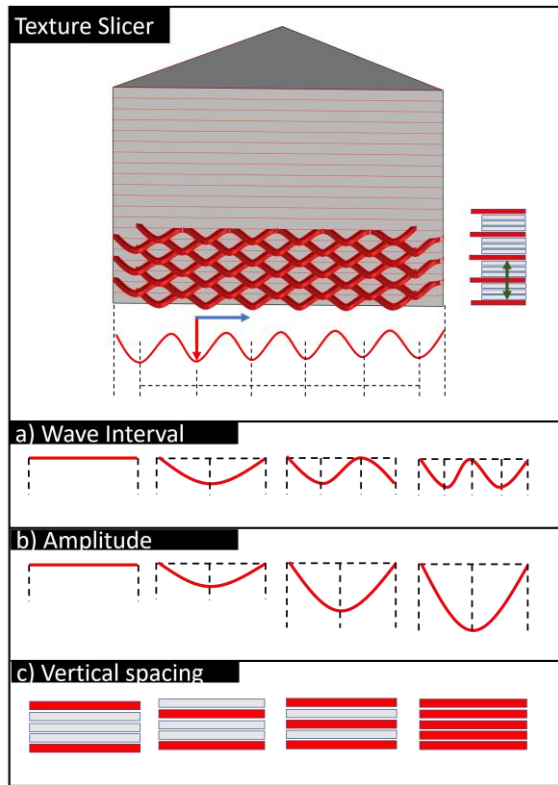


Figure 1 Slicer with computational texture generator

3. TextureSlicer

TextureSlicer is a computational slicer for FDM printers. It converts a geometry to a customizable g-code, generating multiple textures from a single geometry source. Once the geometry is introduced to our slicer, which is scripted in Grasshopper Rhino, it generates 3D printing paths. To create textured surface, geometry cross-sections are converted into curvature paths oscillating around them. *TextureSlicer* takes three parameters to generate multiple texture patterns: a) wave-interval, b) amplitude, and c) vertical spacing.

a) **Wave Interval** is the distance between the two consecutive waves (Figure 1.A) The shorter gap between the waves increases the frequency, resulting in steep waves given the same amplitude. A larger interval results in fewer waves, creating smoother surface finishing.

b) **Amplitude** is the distance between a geometry's outer edge and the oscillation pattern's furthest point (Figure 1.B). It controls the steepness of concave curvature, creating evident surface texture on a 3D-printed object.

c) **Vertical Spacing** is the distance between the two consecutive texture layers (Figure 1.C). The non-texture layer follows the original geometry printing path. Having these

non-texture layers in between the texture layers, the print path of the texture layer ends up printing in midair, resulting in the materials falling downward due to its gravity and absence of support underneath.

4. Texture Exploration

In this section, we 3D printed 20 texture samples of all possible combinations of four intervals (1 mm to 4mm with 1mm increment) and five amplitudes (1mm to 5mm with 1mm increment) without any vertical spacings. Additionally, we printed 20 identical texture samples with 1mm vertical spacing to explore their characteristics.

The texture pattern and g-code were generated in Rhino Grasshopper with our custom C# script and printed using an e3D tool changer (only with a single nozzle) with PLA filament under 200 degree Celsius. A g-code of this geometry can be computed in up to 61 milliseconds, while parameter injection can take up to 22 milliseconds.

In Figure 2.A, printed samples show that the more considerable interval distance (4mm interval) creates a more vivid texture pattern than the ones with a shorter interval (1mm). On the other hand, amplitude also plays a significant parameter to augment the texture effect regardless of the interval distances. While the amplitude magnifies the textured effect on all interval samples, it increases the overall size of the geometry, which designers might not desire.

Figure 2.B shows all 20 samples with every possible interval and amplitude combination with 1mm vertical spacing. While the printing parameters (wave intervals and amplitude) are identical to those in Figure 2.A, the samples with vertical spacing create distinct textures and patterns compared to the previous samples (Figure 2.A). This is due to the lack of support underneath the printed texture-path, and materials are sagging of its weight and gravity, therefore creating a weaving effect. Another notable feature from Figure 2.B samples is that their amplitude does not increase the overall size of the geometry compared to the one without the vertical spacing (Figure 2.A), as the texture patterns are falling downwards.

In total we printed 40 texture samples to understand how the three parameters generate various texture effects and characteristics.

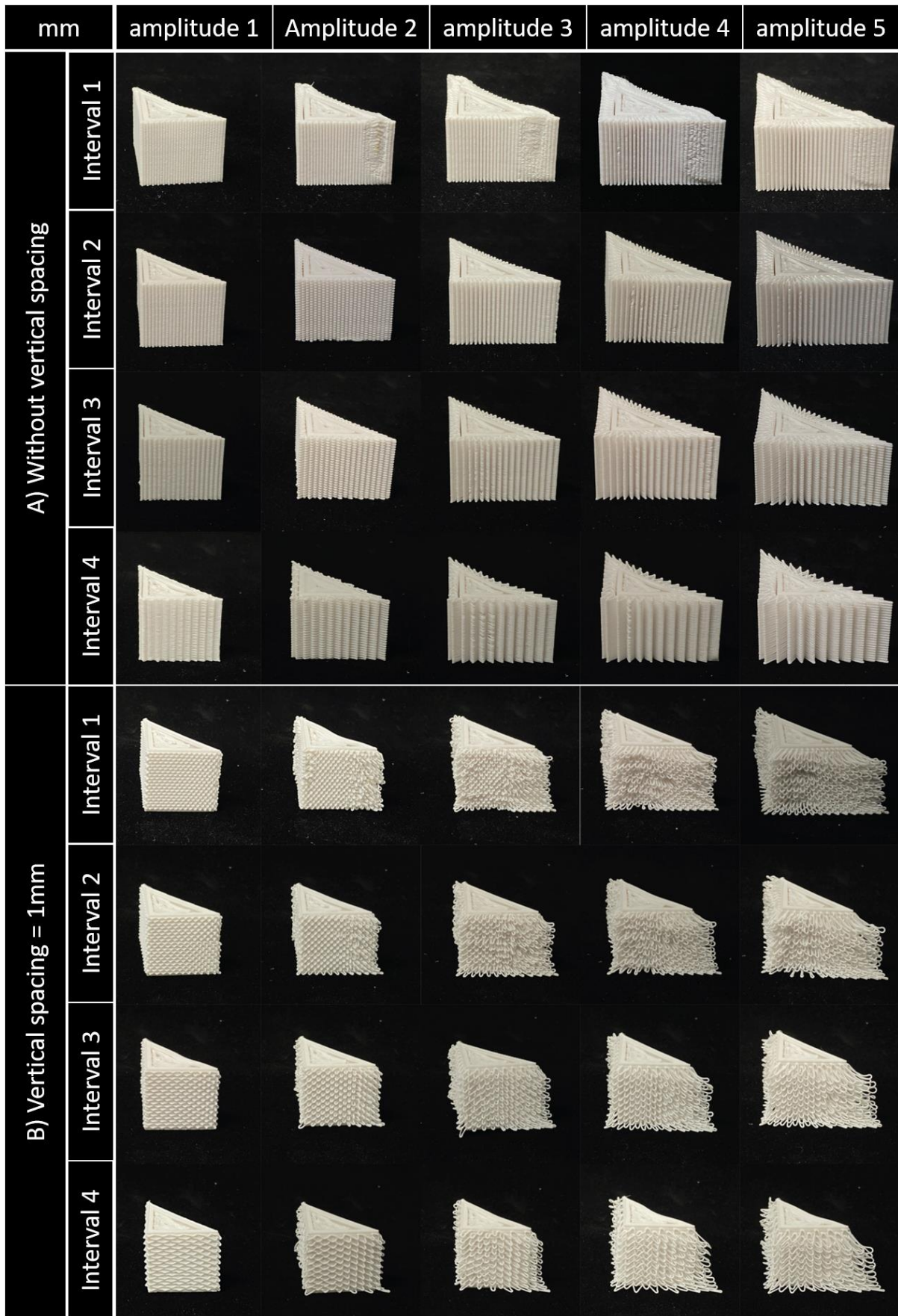


Figure 2 A) 20 samples with all combinations of interval distances and amplitudes without vertical spacings, B) 20 identical samples with vertical spacing 1mm

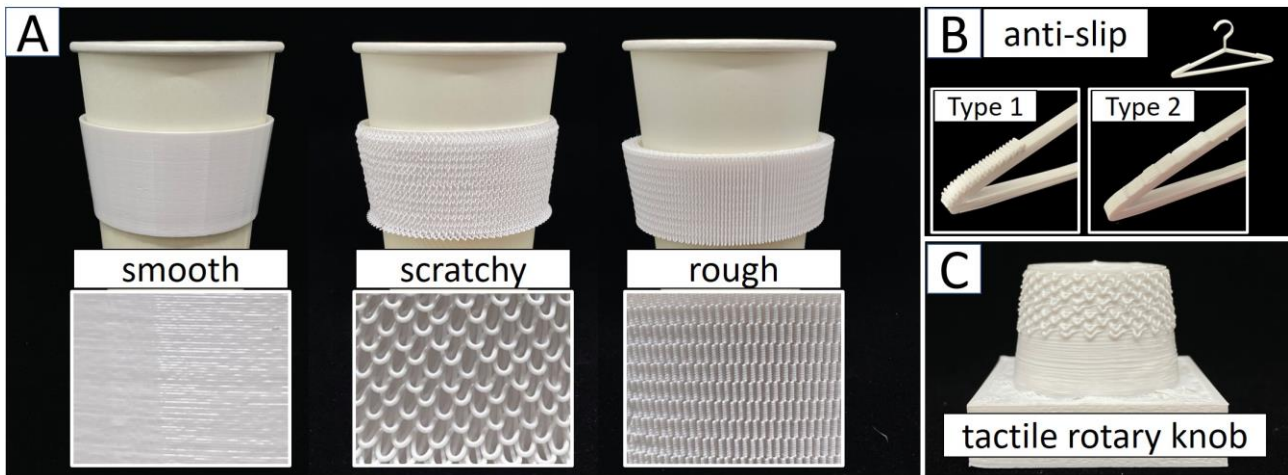


Figure 3 A) 3D printed cup holder with three different texture patterns: smooth, scratchy and rough. They are all generated through *TextureSlicer* from a single geometry., B) A cloth hanger with a rough texture on its shoulders to prevent the cloth from slipping. Users can change the frequency of the texture pattern to control the friction, C) A tactile rotary knob for tangible user interface, various tactile feedback from different textures allows users to distinguish multiple tangible interfaces from each other.

5. Applications

Previous works [7,8,9] have generated different textures for a cup holder by re-modelling the entire geometry for each texture pattern. However, *TextureSlicer* can obtain different surface finishing and features that can be computed on g-code without re-modelling the original geometry.

TextureSlicer will eventually allow researchers as in [7,8,9] to quickly explore different texture patterns that affect taste perceptions when drinking beverages. Figure 3.A shows three cup holders that are 3D printed from a single geometry source, having different texture parameters. For instance, the smooth pattern has zero interval and zero amplitude distance, while the scratchy texture is formed through 3mm interval, 3mm aptitude and 1 mm vertical spacing. The cup holder with a rough finish is created through a 3mm interval distance, 3mm amplitude and no vertical spacing.

Figure 3.B shows a hanger with resistive texture on its shoulders. *TextureSlicer* allows users to modify the texture of a surface selectively. Our system will enable users to quickly generate various resistive patterns such as in type 1 and type 2, for different cloth weights without changing the hanger geometry.

Figure 3.C demonstrates a tangible rotary knob with tactile feedback generated through

TextureSlicer. Our system allows users to design tangible interfaces with multiple tactile feedback that differ from each other. Users can obtain this process without repetitively 3D modelling geometries for each tactile feedback, but from a single knob model to numerous 3D printed interfaces with distinct tactile feedback.

6. Conclusion

In this paper, we proposed a novel approach on 3D printing various textured-object without geometrically modelling the g-code itself but computationally modifying the g-code of a 3D printer. Our system allows users to rapidly generate print-ready files with different textures by adjusting three key parameters that render distinct features from each other. Since the current system enables printing a textured surface on the object sides only, we plan to continue the research by developing textured surfaces on top of a geometry with extrusion rate and printing speed. We also plan to conduct a study to explore how people perceive those textures.

Acknowledgement

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. 2020R1A2C1012233).

References

- [1] Haruki Takahashi, Parinya Punpongsanon, and Jeeun Kim. 2020. Programmable Filament: Printed Filaments for Multi-material 3D Printing. In Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology (UIST '20). Association for Computing Machinery, New York, NY, USA, 1209–1221. <https://doi.org/10.1145/3379337.3415863>
- [2] Stefanie Mueller, Sangha Im, Serafima Gurevich, Alexander Teibrich, Lisa Pfisterer, François Guimbretière, and Patrick Baudisch. 2014. WirePrint: 3D printed previews for fast prototyping. In Proceedings of the 27th annual ACM symposium on User interface software and technology (UIST '14). Association for Computing Machinery, New York, NY, USA, 273–280. <https://doi.org/10.1145/2642918.2647359>
- [3] Franklin Pezutti-Dyer and Leah Buechley. 2022. Extruder-Turtle: A Library for 3D Printing Delicate, Textured, and Flexible Objects. In Sixteenth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '22). Association for Computing Machinery, New York, NY, USA, Article 6, 1–9. <https://doi.org/10.1145/3490149.3501312>
- [4] Jasper Tran O'Leary, Eunice Jun, and Nadya Peek. 2022. Improving Programming for Exploratory Digital Fabrication with Inline Machine Control and Styled Toolpath Visualizations. In Proceedings of the 7th Annual ACM Symposium on Computational Fabrication (SCF '22). Association for Computing Machinery, New York, NY, USA, Article 8, 1–12. <https://doi.org/10.1145/3559400.3561998>
- [5] Jifei Ou, Gershon Dublon, Chin-Yi Cheng, Felix Heibeck, Karl Willis, and Hiroshi Ishii. 2016. Cillia: 3D Printed Micro-Pillar Structures for Surface Texture, Actuation and Sensing. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16). Association for Computing Machinery, New York, NY, USA, 5753–5764. <https://doi.org/10.1145/2858036.2858257>
- [6] Lingyun Sun, Jiaji Li, Yu Chen, Yue Yang, Ye Tao, Guanyun Wang, and Lining Yao. 2020. : A Shape-Changing Fabrication Method for 3D Surfaces with Texture. In Extende4DTextured Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems (CHI EA '20). Association for Computing Machinery, New York, NY, USA, 1–7. <https://doi.org/10.1145/3334480.3383053>
- [7] van Rompay, T. J. L., Finger, F., Saakes, D., & Fenko, A. (2017). “See me, feel me”: Effects of 3D-printed surface patterns on beverage evaluation. *Food Quality and Preference*, 62. <https://doi.org/10.1016/j.foodqual.2016.12.002>
- [8] van Rompay, T., van Ooijen, I., Groothedde, S., & Saakes, D. (2021). (Not to be taken) with a grain of salt: Enhancing perceived saltiness by 3D-printed surface textures. *Food Quality and Preference*, 93. <https://doi.org/10.1016/j.foodqual.2021.104279>
- [9] van Rompay, T. J. L., Kramer, L. M., & Saakes, D. (2018). The sweetest punch: Effects of 3D-printed surface textures and graphic design on ice-cream evaluation. *Food Quality and Preference*, 68. <https://doi.org/10.1016/j.foodqual.2018.02.015>