FabricFaces: Combining Textiles and 3D Printing for Maker-Friendly Folding-Based Assembly

Adrian Wagner
wagner@cs.rwth-aachen.de
RWTH Aachen University
Aachen, Germany

Paul Miles Preuschoff
paul.preuschoff@rwth-aachen.de
RWTH Aachen University
Aachen, Germany

Philipp Wacker
wacker@cs.rwth-aachen.de
RWTH Aachen University
Aachen, Germany

Simon Voelker
voelker@cs.rwth-aachen.de
RWTH Aachen University
Aachen, Germany

Jan Borchers
borchers@cs.rwth-aachen.de
RWTH Aachen University
Aachen, Germany

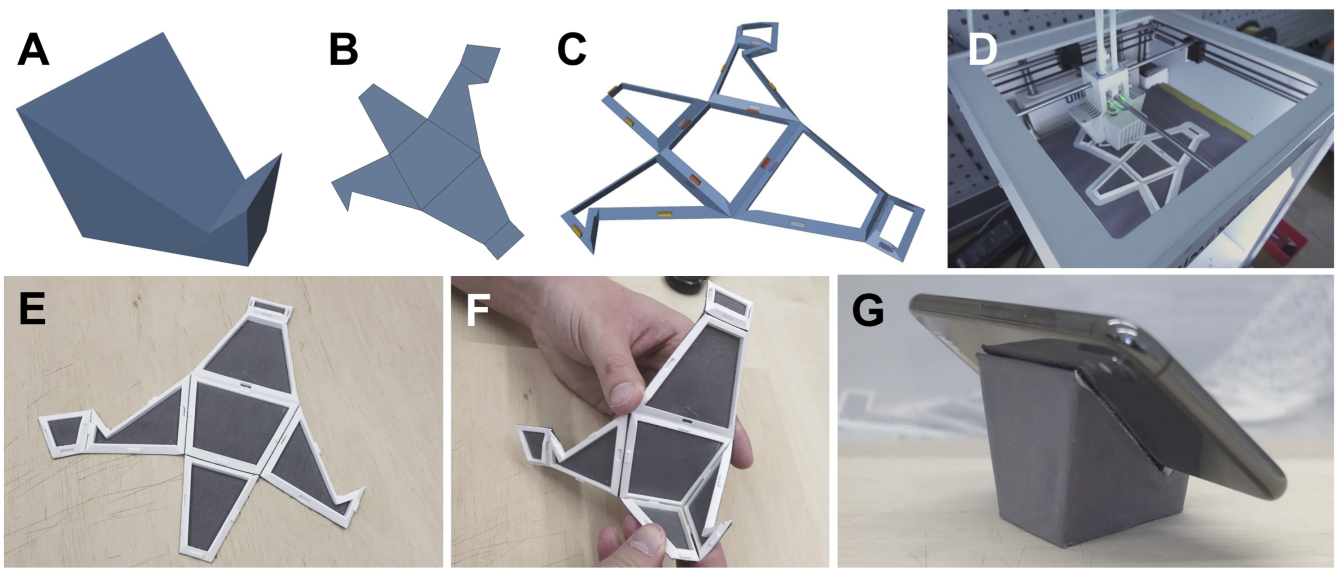

Figure 1: The steps of the FabricFaces workflow, using a smartphone stand as an example: Our tool takes a user-provided virtual model (A), unfolds it into a flat geometry (B), and generates a new 3D-printable model of frames with connectors for assembly (C). The user then 3D prints these frames onto fabric (D), cuts away any excess fabric (E), and assembles the object by folding (F). The final result is an object with fabric-covered faces, like this smartphone stand (G). This workflow makes it easy to create 3D objects with fabric surfaces even with just a hobbyist-level 3D printer. By merging two distinct crafting workflows, it opens up an exciting new design space of textile-covered 3D-printed objects.

ABSTRACT

We introduce a Personal Fabrication workflow to easily create feature-rich 3D objects with textile-covered surfaces. Our approach unfolds a 3D model into a series of flat frames with connectors, which are then 3D-printed onto a piece of fabric, and folded manually into the shape of the original model. This opens up an accessible way to incorporate established 2D textile workflows, such as embroidery, using color patterns, and combining different fabrics, when creating 3D objects. FabricFaces objects can also be flattened again easily for transport and storage. We provide an open-source plugin for the common 3D tool Blender. It enables a one-click workflow to turn a user-provided model into 3D printer instructions, textile cut patterns, and connector support. Generated frames can be refined quickly and iteratively through previews and extensive options for manual intervention. We present example objects illustrating a variety of use cases.

CHI EA '23, April 23–28, 2023, Hamburg, Germany

ACM ISBN 978-1-4503-9422-2/23/04.
https://doi.org/10.1145/3544549.3585854

CCS CONCEPTS

• **Human-centered computing → Human computer interaction (HCI)**.

KEYWORDS

Fabrication, 3D Printing, Textiles

ACM Reference Format:

Adrian Wagner, Paul Miles Preuschoff, Philipp Wacker, Simon Voelker, and Jan Borchers. 2023. FabricFaces: Combining Textiles and 3D Printing for Maker-Friendly Folding-Based Assembly. In *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems (CHI EA '23), April 23–28, 2023, Hamburg, Germany*. ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3544549.3585854

1 INTRODUCTION

The rise of 3D printing in Personal Fabrication required not only printers to become affordable enough for widespread adoption, but also the printing process to be adapted and simplified for a more heterogenous and non-professional user base that includes beginners. Additionally, the printing process rarely happens in isolation, but connects to other steps of larger, interconnected Personal Fabrication workflows. Frequent switches between different tasks, tools, and even users occur often [5]. The resulting standard process for makers typically entails 3D modeling, STL export, slicing to generate G-code, printing, and post-processing. This standardization ensures interoperability between different software tools and printers, and lets users choose their preferred tools and still collaborate. But standardization restricts creative expression through constraints such as limited colors, print sizes, and material choices.

Textile design and fabrication workflows, on the other hand, have evolved for centuries to support both expressiveness and ease of use with simple tools. They offer a large variety of options for the color and feel of surfaces, can be enhanced through textile printing, embroidery, or mixing fabrics, support collaboration, and integrate well into larger workflows. Unlike 3D prints, textiles offer a large range of material flexibility and stretchability, and can be bent without breaking, but also provide structural integrity and rigidity when used in compound materials.

We introduce *FabricFaces,* a maker-friendly Personal Fabrication workflow that combines 3D printing with textiles, benefitting from the advantages of both workflows. Our workflow automatically unfolds a user-provided 3D model into a flat surface. We then turn this flat surface into a series of 3D-printable frames. The frames feature angled edges and automatically generated connectors for joining them at the required angle. This disassembled, flattened wireframe can then be printed directly onto a fabric surface attached to the print bed of an unmodified hobbyist-level fused deposition modeling (FDM) printer. After cutting out this flattened structure from the fabric, the user can fold up and snap together the fabric-covered wireframe to create the original 3D object.

To support our proposed FabricFaces workflow and evaluate our approach with users, we created an open-source software tool[1] as a plugin for the common open-source 3D design software Blender [2].

Our key contributions, therefore, are *FabricFaces*, a new maker-friendly Personal Fabrication workflow that combines textiles and 3D printing to let users create textile-covered 3D objects quickly and easily with readily available tools and materials, and a software tool that implements our workflow to enable further exploration and experimentation within the standard 3D printing workflow. To support the validity of our proposed FabricFaces approach, we provide a number of application examples and interview findings from a workshop with users of our workflow and tool.

2 RELATED WORK

Our workflow builds on previous work in fabrication through assembly with connectors and folding, wireframe structures, and fabric embeds in 3D printing.

Since standard materials like wood and metal usually come as flat sheets, it is feasible to construct 3D objects by assembling them from interlocking 2D parts. Schwartzburg and Pauly [20], and McCrae et al. [12] showed how flat interlocking pieces with cutouts create a representation of the silhouette of a 3D object. Similarly, Kyub [3] generates objects with volume from interlocking parts that can be produced on laser cutters, while LamiFold [10] lets users create physical objects with built-in mechanisms by stacking flat cutouts but also relying on adhesives for structure.

By using one connected piece of flat material, interlocking and adhesives can be avoided, and 2D parts assembled into 3D objects by *rigid folding* along seams [9, 21]. Folding can be applied automatically in numerous ways, e.g., by melting plastic held at an angle [14] with a laser cutter, or by cutting stripes of foldable panel structures [1]. This workflow can also be translated to other devices, such as computer numerical control (CNC) machines, using a limited number of differently angled cutting heads [15].

While assembly often aims at obtaining closed surfaces, utilizing wireframe structures enables faster iteration. For FabricFaces, we chose to only print the outer edges of faces, similar to Mueller et al. [13] or the more user-driven approach presented by Peng et al. [17]. In contrast to these works, however, we compute an unfolded wireframe and print it flattened, instead of having to print the wireframe itself in 3D.

While it is possible to create bendable fabric-like structures with 3D printing only, those methods are restricted to special printing setups, like contructing pillars on the print bed [22], or dedicated printer assemblies for electrospinning [18]. We are interested in the new opportunities that arise from rigid folding by covering the surface of a wireframe-like 3D print with fabric, a concept thoroughly examined by Rivera et al. [19].

3 THE FABRICFACES WORKFLOW

The simplified view of standard 3D printing workflow entails ideation, 3D modeling, preparation for slicing, the slicing operation, printing, and post-processing. We based our workflow on this simplified view to make FabricFaces available to a broad audience. Furthermore, we focused on a process using the standard entry-level 3D printing technology most widely used in the DIY community today, fused deposition modeling (FDM).

[1] https://hci.rwth-aachen.de/fabricfaces
[2] https://blender.org

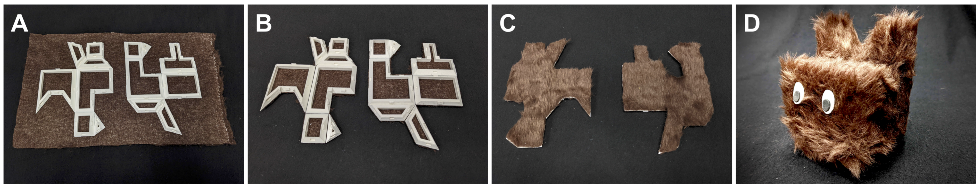

Figure 2: An example of using fluffy material in an object representing a cat head. (A) shows the assembly of two disconnected parts on fabric with glue. (B) shows the two parts cut out from the inside and (C) on the outside. (D) is the final assembled object.

To illustrate how our FabricFaces workflow extends this standard process, we first provide a user scenario for the simplest use case of FabricFaces (Fig. 1):

> *Sally, a maker, wants to create a phone stand with a soft but grippy surface she can take with her easily. Applying such a surface texture to a 3D print is difficult, but achieving it with fabrics using FabricFaces is easy: She imports a model she found on Thingiverse[3] into our software tool, which automatically cuts and unfolds the model into frames with connectors she can print directly onto the fabric of her choice. She fastens her fabric to the print bed, slices the newly generated frames model, and prints it like any other object. After cutting off the excess fabric around the frames and folding them up until the connectors latch, she holds her new phone stand in her hands. Its surfaces are covered with the fabric she picked, giving it a nice feel, and allowing it to unfold easily after use to take with her.*

In this most basic version, the workflow enables user-provided models to be processed automatically so that they can be printed directly onto fabric. Surface characteristics that would be difficult to achieve using 3D printing can be achieved conveniently by using fabric. These include intricate texturing and coloring, water absorption [18], and flexibility. Additionally, assembly by folding is a reversible process. For transport or storage, objects can be unfolded again to lay flat and take up less space.

If any characteristics of the fabric chosen prohibit fastening it tightly onto the print bed, the frames can also be printed directly on the print bed as usual and glued onto the fabric later. We show this variation in Fig. 2.

Advanced users can manually adjust our automatic processes for deeper design opportunities, seen in Fig. 3. Changing connector types and marking connectors for omission enables control of local frame bonding. Together with the ability to manually mark edges for mandatory cutting, we can achieve moving parts on an otherwise simple 3D model through our workflow. As an example, Fig. 4 shows a jewelry case with a moving lid. The hinge is realized by introducing an edge without connectors and closes with low-strength connectors on the opposite side. This jewelry case also showcases one of the artistic possibilities when working with fabric. By not cutting away the fabric around the frames on all edges, the resulting folds provide a distinct look.

We expose manual scale and size settings for the frames and all parts they consist of, including connectors between frames. This can be used to create objects far larger than normally possible on a given 3D printer. To illustrate this, we created a standing lamp, shown in Fig. 5, which exceeds the print volume of our 3D printer many times, by printing the frames individually (possibly in parallel on multiple printers). Here, we chose to arrange and glue the frames onto one single, large piece of fabric afterwards, achieving a seamless look, and aiding in assembly. To hold electronics in place, we exported one generated frame in STL format, and added a hook in our 3D modeling program of choice before printing it. For user input, we sewed traces of conductive yarn into the top, a standard textile workflow. We chose translucent white fabric to support the lampshade aesthetics.

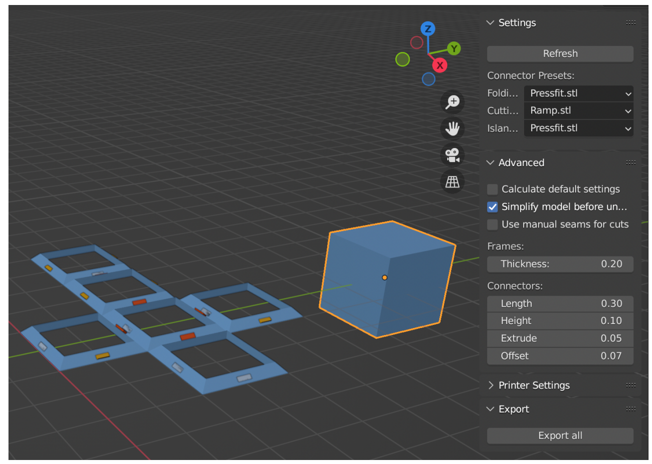

Figure 3: The interface of our plugin in Blender. The automatic process can be tuned manually via the Blender tool shelf on the right.

4 SOFTWARE ARCHITECTURE AND IMPLEMENTATION

Internally, the FabricFaces workflow is split into four distinct phases:

(1) Unfolding an arbitrary user-provided object, shown in Fig. 1a,b.
(2) Generating a frame with angled sides populated with connectors, shown in Fig. 1c.
(3) 3D printing onto textile, shown in Fig. 1d,e.
(4) Assembly by folding and press fitting the connectors, shown in Fig. 1f.

Below, we provide an overview of the resulting software architecture with some implementation details.

[3]https://www.thingiverse.com/

Figure 4: A jewelry case with a lid that can be opened, and with fabric folds as aesthetic elements: (A) The unfolded case with the additional fabric. (B) The closed case and its aesthetics. (C) The case with the lid open; red arrows mark the hinge without connectors and the connector that helps hold the lid closed.

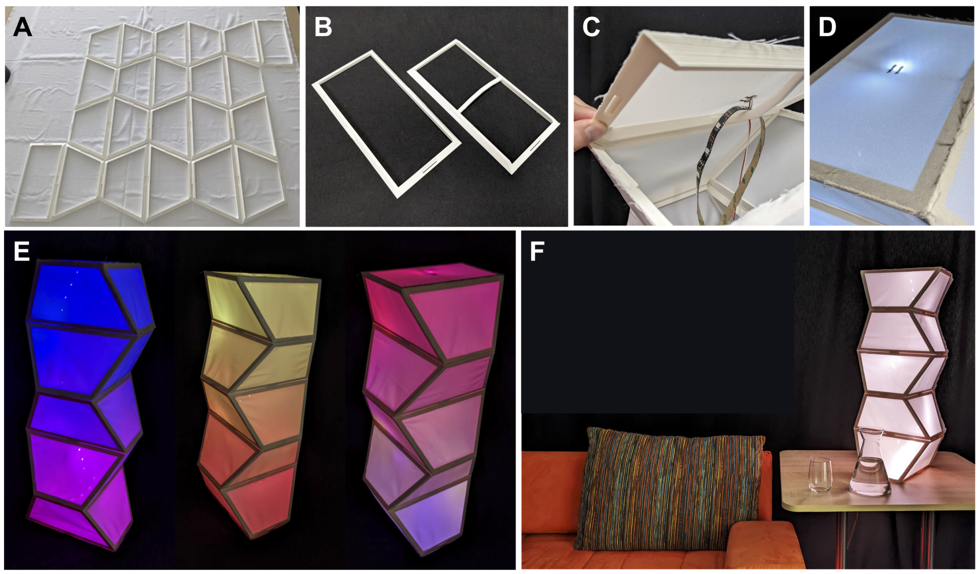

Figure 5: A large table lamp with functionality added through embedded electronics: (A) The individual frames arranged on fabric for folding. (B) The generic top frame (left), and the manually augmented top frame (right). It holds the electronics in (C). (D) The sewn-on switch using conductive fabric. (E) The lamp shown from different angles, displaying a variety of light patterns. (F) An application scenario.

4.1 Preprocessing

Our workflow needs to account for a wide range of properties of the user-provided 3D model that may interfere with our process. To this end, we implemented an automatic preprocessing step, freeing the user of the burden of applying our model restrictions.

In a virtual representation, faces consisting of arbitrary polygons can be non-planar. Our approach needs flat-lying faces to be printable directly onto fabric. We chose to planarize faces by aligning all corners on one average plane.

Models are often highly tessellated, especially if exported from computer aided design (CAD) software. However, frame sizes are limited by the 3D printer's minimal layer height and nozzle size, which defines a minimum face size, as the frame of each face has to be packed inside its footprint. Even if we could print arbitrarily accurately, we would still be restricted by human dexterity (and patience) during assembly. We thus chose to automatically merge all faces that meet at close to 180 degrees, independent of their size. Advanced users can skip this step, risking unserviceable output for greater flexibility.

4.2 Unfolding

After ensuring that the input fulfills the requirements, it is transformed onto a flat plane. We aim to align as many faces as possible to each other on their shared edge to help with folding during assembly. Our approach, therefore, resembles an unfolding of rigid parts that are connected by a hinge on their shared edges. However, if overlaps occur in the unfolding, this would mean that two faces partially occupy the same space on the unfolding plane. Since in some cases, finding one connected unfolding without overlaps is impossible, models may be unfolded into disconnected parts if needed, each consisting of multiple connected faces.

We adapted the smart unfolding algorithm introduced by Muntoni et al. [15] to our unfolding step. It uses an iterative approach to reduce the number of disconnected parts, and have them extend

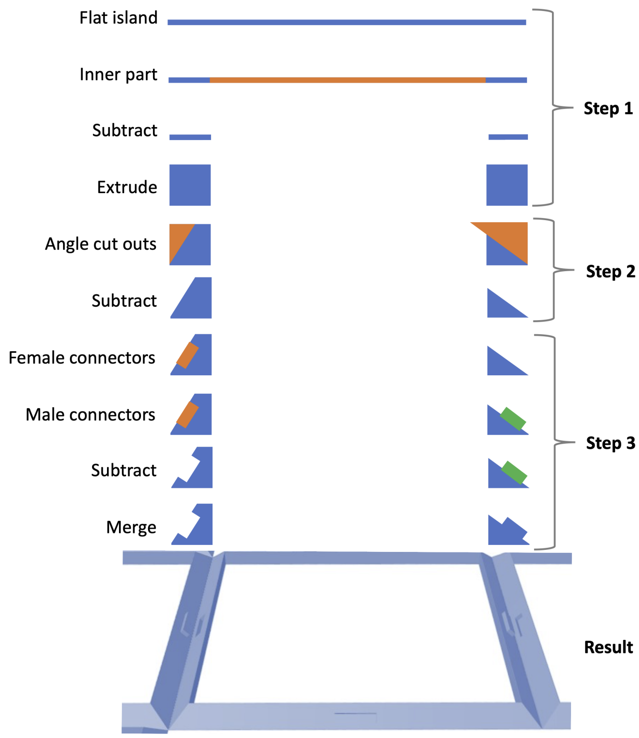

Figure 6: Frames are computed by generating borders from an unfolding, converting them to borders and extrusion (Step 1). Angled cut-outs are subtracted to create frames that fold neatly onto each other (Step 2). These angled parts are populated with male and female connectors (Step 3).

mostly equally into all directions. It achieves this by starting with a seed face and then unfolding the connected faces in a breadth-first approach. If that results in overlaps and consequently disconnected parts, the number of seed faces is increased, and multiple seed faces are unfolded simultaneously. This repeats until the unfolding stops generating more disconnected parts than the amount of starting seed faces in the specific iteration. In our implementation, we chose the largest face as the first seed face because it is often also the face with the most connected faces. We further adapted the algorithm to our needs by adapting the rules governing when further unfolding of an island is considered possible from CNC machines to 3D printers. We also added the ability to mark edges as mandatory cutting areas, giving manual control over island growth and possible folding patterns. Overall, this supports a straightforward assembly process and enables manufacturing in smaller workspaces. For a general overview of the algorithm, we refer to [15].

4.3 Frame Generation

Theoretically, the edges of the unfolded mesh could already be printed on fabric, resulting in papercraft-like folding patterns. While this could be folded, it would lack guidance on folding angles and not hold the resulting shape together. Therefore, we compute *frames,* as shown in Fig. 6.

Every frame encapsulates one face and serves as a border that the textile adheres to in order to cover the corresponding face. The frame is extended upwards in a wedge-like shape to strengthen the object and guide users to the right angle during assembly. They only have to move faces toward each other until movement is restricted.

We populate the frames with *connectors* to enable the assembly of sturdy objects. Irregular press-fit connectors are an obvious choice here [11]. However, our approach accepts connectors with arbitrary shapes, opening up the design space. More complex connectors can help in specific situations, and experimentation is encouraged.

5 PRELIMINARY VALIDATION

As a preliminary validation, we conducted a workshop with seven participants (two female) with prior knowledge of CAD tools and 3D printing to understand how they accept the FabricFaces workflow, and to see what objects they would come up with if given complete design freedom.

5.1 Methodology

The workshop was held remotely over the time of one week, with participation spread out over that period. We started with a 20-min introduction of the supporting tool. Using Zoom [4] screen sharing, we demonstrated the features of the tool and how to install it. We then entered the design phase, stopped the Zoom call, and participants worked by themselves for about 2 hours. Participants created 3D objects in a software of their choice and used our tool to generate the frames from them and configure parameters with a live preview. After the workshop, participants sent us the 3D model files of the objects they had designed.

Based on the workshop results, we conducted unstructured follow-up interviews with participants. We asked questions about the type of objects they created, the iterative process of their creation, and the changes that were planned but not implemented due to time or feature constraints. Answers were logged by hand and combined with questions we received via Slack [5] during the individual work period for additional insights. After the interview, participants could request their models to be printed, as the remote workshop did not allow this directly.

5.2 Results

We received at least one object from every participant (Fig. 7 shows examples). From the unstructured interviews, we gathered that all participants tried designing multiple objects, but some had problems with the limitations of folding construction: "I had problems finding good settings. [...] I used the plugin for iterative design" (P2).

We received objects with different intentions behind the design process, like purely functional objects: "I used existing objects from private projects" (P1); "I took measures of a real object I own" (P2), or aesthetically pleasing objects: "I wanted to create functional objects, but ended up with good-looking ones" (P5). Participants also explored the design space thoroughly: "Holes end up disassembling the object into many frames" (P5); "I created the jewelry box to figure out how roundness works with folding" (P3); "I know the design should work, but I did not manage it in time" (P6); "I would

[4]https://zoom.us
[5]https://slack.com

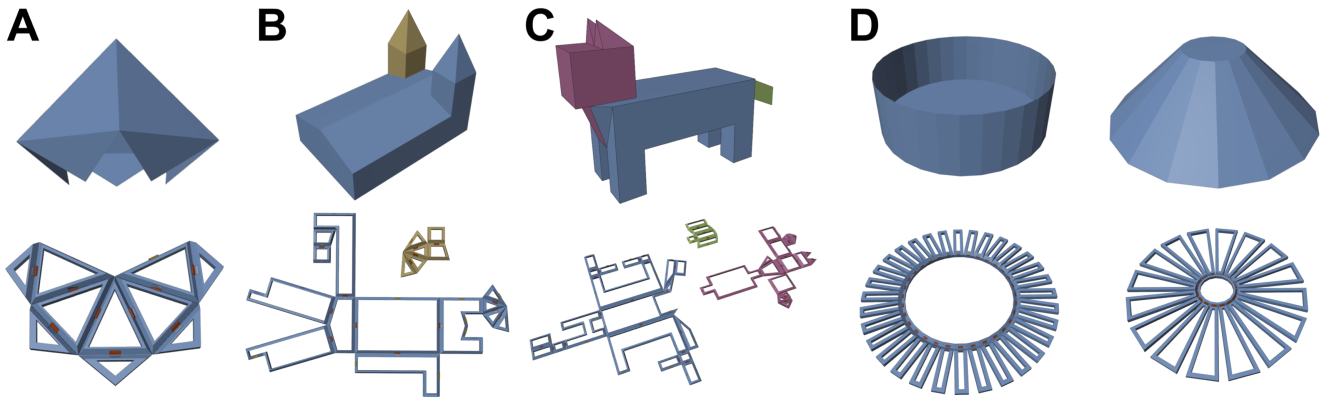

Figure 7: We present a few examples of objects generated with FabricFaces and their unfolded frames. These objects are (A) "Lampshade", (B) "Church", (C) "Cat" and (D) "Jewelry Box". While object (A) was one of our examples, the other objects were generated by participants in a workshop.

like to test how sturdy such an object would be" (P1); and asked for more possibilities: "Do you think I can also print on tracing paper?" (P3); "How big can I make an object?" (P4).

6 LIMITATIONS AND FUTURE WORK

While we designed the FabricFaces workflow to accept arbitrary objects, we were not able to account for arbitrary complexity. This is partially alleviated by a pre-processing step, which in turn may generate unwanted changes to the original model. To balance processability and shape retention, we expose advanced settings and propose different connectors. Still, objects are limited by machine capabilities like 3D printer nozzle size, and by human dexterity during assembly. Our implementation only aims at producing well-formed output within these confines.

While we created a small library of connector types with recommendations on usage, we aim to evaluate connectors in more detail. We expect that further research into more application areas could extend this library further.

While our workflow stands on its own without the accompanying implementation, we have to acknowledge the additional restrictions put on the workflow through our tool. For our implementation, we settled on Smart Unfolding from Muntoni et al. [15], with simplified machining as a goal. One candidate to simplify the assembly process for a user may be the algorithm in [8] that aims to create unfoldings that refold easily.

Automatic self-folding is another promising direction. For 2.5D structures, Goudswaard et al. [4] proposes relief-like structures that raise themselves. This approach prints directly onto pre-stretched fabric, but, unlike FabricFaces, uses stretchy fabric. De-stretching this fabric starts the self-assembly. Curved folding allows for standard 3D prints to self-assemble when using thermoplastic composites, as presented by An et al. [2]. Muthukumarana et al. [16] control movement of rigid 3D prints on fabric by embedding additional materials that retract in length through heat. Additionally, changing the unfolding algorithm to allow for linear movement during assembly without the frames colliding with each other, introduced by Hao et al. [7], helps achieve this goal.

While we used conductive yarn on the textile surface in Fig. 5, we applied the yarn manually. Automatically routing conducting traces, and using embroidery to cover our surfaces with e-textiles that can sense gestures similar to Hamdan et al. [6] and Goudswaard et al. [4], is one of the next challenges to tackle.

Our preliminary evaluation showed interest in the workflow and the possibilities opened up with live exploration using our tools. Further evaluation is necessary with regard to integrating this workflow into 3D printing software as well as the opportunities opened up by the possible parameter spaces for advanced users.

7 CONCLUSION

This paper introduced FabricFaces, a new workflow to easily design and assemble 3D prints with textile surfaces. This workflow allows users to apply color, surface finishes, and different textures to 3D prints more easily, by leveraging the combined workflows of textile craft and 3D printing, which have each evolved sophisticated design processes in different application areas. We created a tool to implement the workflow to preliminarily evaluate our process in an online workshop. The results of this workshop show FabricFaces was not only useful to design particular textile-covered objects, but that it also sparked ideas of interesting new designs and functionality. We hope that our FabricFaces workflow and supporting tool help push the envelope of embedded 3D printing, and that researchers and designers enjoy further exploring the exciting new design space of textile-covered 3D printed objects that FabricFaces unlocks.

ACKNOWLEDGMENTS

We thank Sam Mattiussi and Robert Rudnev for their help with implementing and testing FabricFaces. This work was funded in part by the German B-IT Foundation.

REFERENCES

[1] Ergun Akleman, Shenyao Ke, You Wu, Negar Kalantar, AliReza Borhani, and Jianer Chen. 2016. Construction with physical version of quad-edge data structures. *Computers & Graphics* 58 (Aug. 2016), 172–183. https://doi.org/10.1016/j.cag.2016.05.008

[2] Byoungkwon An, Ye Tao, Jianzhe Gu, Tingyu Cheng, Xiang 'Anthony' Chen, Xiaoxiao Zhang, Wei Zhao, Youngwook Do, Shigeo Takahashi, Hsiang-Yun Wu, Teng Zhang, and Lining Yao. 2018. Thermorph: Democratizing 4D Printing of Self-Folding Materials and Interfaces. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3173574.3173834

[3] Patrick Baudisch, Arthur Silber, Yannis Kommana, Milan Gruner, Ludwig Wall, Kevin Reuss, Lukas Heilman, Robert Kovacs, Daniel Rechlitz, and Thijs Roumen. 2019. Kyub: A 3D Editor for Modeling Sturdy Laser-Cut Objects. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. ACM, Glasgow Scotland Uk, 1–12. https://doi.org/10.1145/3290605.3300796

[4] Maas Goudswaard, Abel Abraham, Bruna Goveia da Rocha, Kristina Andersen, and Rong-Hao Liang. 2020. FabriClick: Interweaving Pushbuttons into Fabrics Using 3D Printing and Digital Embroidery. In *Proceedings of the 2020 ACM Designing Interactive Systems Conference* (Eindhoven, Netherlands) *(DIS '20)*. Association for Computing Machinery, New York, NY, USA, 379–393. https://doi.org/10.1145/3357236.3395569

[5] Bruna Goveia da Rocha, Johannes M. L. van der Kolk, and Kristina Andersen. 2021. Exquisite Fabrication: Exploring Turn-Taking between Designers and Digital Fabrication Machines. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 434, 9 pages. https://doi.org/10.1145/3411764.3445236

[6] Nur Al-huda Hamdan, Simon Voelker, and Jan Borchers. 2018. Sketch&Stitch: Interactive Embroidery for E-Textiles. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3173574.3173656

[7] Yue Hao, Yun-hyeong Kim, and Jyh-Ming Lien. 2018. Synthesis of Fast and Collision-Free Folding of Polyhedral Nets. In *Proceedings of the 2nd ACM Symposium on Computational Fabrication* (Cambridge, Massachusetts) *(SCF '18)*. Association for Computing Machinery, New York, NY, USA, Article 2, 10 pages. https://doi.org/10.1145/3213512.3213517

[8] Yue Hao, Yun-hyeong Kim, Zhonghua Xi, and Jyh-Ming Lien. 2018. Creating Foldable Polyhedral Nets Using Evolution Control. In *Robotics: Science and Systems XIV*. Robotics: Science and Systems Foundation. https://doi.org/10.15607/RSS.2018.XIV.007

[9] Arthur Lebée. 2015. From Folds to Structures, a Review. *International Journal of Space Structures* 30, 2 (2015), 55–74. https://doi.org/10.1260/0266-3511.30.2.55 arXiv:https://doi.org/10.1260/0266-3511.30.2.55

[10] Danny Leen, Nadya Peek, and Raf Ramakers. 2020. LamiFold: Fabricating Objects with Integrated Mechanisms Using a Laser Cutter Lamination Workflow. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology* (Virtual Event, USA) *(UIST '20)*. Association for Computing Machinery, New York, NY, USA, 304–316. https://doi.org/10.1145/3379337.3415885

[11] Linjie Luo, Ilya Baran, Szymon Rusinkiewicz, and Wojciech Matusik. 2012. Chopper: Partitioning Models into 3D-Printable Parts. *ACM Trans. Graph.* 31, 6, Article 129 (nov 2012), 9 pages. https://doi.org/10.1145/2366145.2366148

[12] James McCrae, Nobuyuki Umetani, and Karan Singh. 2014. FlatFitFab: interactive modeling with planar sections. In *Proceedings of the 27th annual ACM symposium on User interface software and technology (UIST '14)*. Association for Computing Machinery, New York, NY, USA, 13–22. https://doi.org/10.1145/2642918.2647388

[13] 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* (Honolulu, Hawaii, USA) *(UIST '14)*. Association for Computing Machinery, New York, NY, USA, 273–280. https://doi.org/10.1145/2642918.2647359

[14] Stefanie Mueller, Bastian Kruck, and Patrick Baudisch. 2013. LaserOrigami: Laser-Cutting 3D Objects. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Paris, France) *(CHI '13)*. Association for Computing Machinery, New York, NY, USA, 2585–2592. https://doi.org/10.1145/2470654.2481358

[15] Alessandro Muntoni, Stefano Nuvoli, Andreas Scalas, Alessandro Tola, Luigi Malomo, and Riccardo Scateni. 2019. Mill and fold: Shape simplification for fabrication. *Computers & Graphics* 80 (May 2019), 17–28. https://doi.org/10.1016/j.cag.2019.03.003

[16] Sachith Muthukumarana, Moritz Alexander Messerschmidt, Denys J.C. Matthies, Jürgen Steimle, Philipp M. Scholl, and Suranga Nanayakkara. 2021. ClothTiles: A Prototyping Platform to Fabricate Customized Actuators on Clothing Using 3D Printing and Shape-Memory Alloys. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 510, 12 pages. https://doi.org/10.1145/3411764.3445613

[17] Huaishu Peng, Jimmy Briggs, Cheng-Yao Wang, Kevin Guo, Joseph Kider, Stefanie Mueller, Patrick Baudisch, and François Guimbretière. 2018. RoMA: Interactive Fabrication with Augmented Reality and a Robotic 3D Printer. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3173574.3174153

[18] Michael L. Rivera and Scott E. Hudson. 2019. Desktop Electrospinning: A Single Extruder 3D Printer for Producing Rigid Plastic and Electrospun Textiles. 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/10.1145/3290605.3300434

[19] Michael L. Rivera, Melissa Moukperian, Daniel Ashbrook, Jennifer Mankoff, and Scott E. Hudson. 2017. Stretching the Bounds of 3D Printing with Embedded Textiles. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. ACM, Denver Colorado USA, 497–508. https://doi.org/10.1145/3025453.3025460

[20] Yuliy Schwartzburg and Mark Pauly. 2013. Fabrication-aware Design with Intersecting Planar Pieces. *Computer Graphics Forum* 32, 2pt3 (2013), 317–326. https://doi.org/10.1111/cgf.12051

[21] Tao Shen and Yukari Nagai. 2017. An Overview of Folding Techniques in Architecture Design. *World Journal of Engineering and Technology* 05, 03 (2017), 12–19. https://doi.org/10.4236/wjet.2017.53B002 Number: 03.

[22] Haruki Takahashi and Jeeeun Kim. 2019. 3D Printed Fabric: Techniques for Design and 3D Weaving Programmable Textiles. 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, 43–51. https://doi.org/10.1145/3332165.3347896