



MASTER THESIS

---

# StackMold

Rapid Prototyping of Functional Multi-Material Objects  
with Selective Levels of Surface Details

---

*Author:*

Tom VALKENEERS

*Supervisor:*

Prof. Dr. Raf RAMAKERS

*A thesis submitted in fulfillment of the requirements  
for the degree of Master Computer Sciences*

*in the*

The Expertise centre for Digital Media (EDM)  
Engineering Interactive Systems

June 6, 2019



## *Acknowledgements*

An enormous thanks to Prof. Dr. Raf RAMAKERS for all his input and support in making this project happen. He provided me with very fast feedback on the work I did, helped with innovative ideas to solve problems and provided me with all the materials and tools that I required. Providing materials and tools is often taken for granted as the duty of the supervisor, but he was proactive in asking if other kinds of material or equipment were needed and when you asked to order a product, it was often ordered the same day. These little details make a big impact in the workflow of a project and me as a student felt supported. The same is true in the way he gave feedback and possible solutions on certain problems or ideas. He knows, and wants to know, in detail how a certain aspect of the software or process functions, which makes his ideas and feedback very practical and on point. Lastly a big thanks for the opportunity to turn this project into a paper. I probably didn't state this enough in person, but I am very glad that I was able to participate in the creation of a paper. I am very happy that I got to do it.

Also a round of applause to Prof. Ph.D. Daniel ASHBROOK and Dany LEEN who both provided feedback on my work despite both being very busy. They also contributed on the paper about this project. A good portion of the pictures from this project are taken and edited by Dany. Thanks to the both of you.

Lastly thanks to Jose Maria TIJERINA MUNOZ, or 'Chema', for helping me out on little things like problems with the lasercutter, casting resin, finding a tool, ... And my girlfriend Charlotte for having to put up with me when I needed to write (I don't like writing at all) or when I had no time while working the paper or around other deadlines.



UHASSELT

# *Abstract*

Faculty Sciences  
Engineering Interactive Systems

Master Computer Sciences

## **StackMold: Rapid Prototyping of Functional Multi-Material Objects with Selective Levels of Surface Details**

by Tom VALKENEERS

We present StackMold, a DIY molding technique to prototype multi-material and multi-colored objects with embedded electronics. The key concept of our approach is a novel multi-stage mold buildup in which casting operations are interleaved with the assembly of the mold to form independent compartments for casting different materials. To build multi-stage molds, we contribute novel algorithms that computationally design and optimize the mold and casting procedure. By default, the multi-stage mold is fabricated in slices using a laser cutter. For regions that require more surface detail, a high-fidelity 3D-printed mold subsection can be incorporated. StackMold is an integrated end-to-end system, supporting all stages of the process: it provides a UI to specify material and detail regions of a 3D object; it generates fabrication files for the molds; and it produces a step-by-step casting instruction manual.



# Contents

<b>Acknowledgements</b>	<b>iii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Importance of multiple material . . . . .	1
1.2 3D multi-material printer . . . . .	3
1.3 Molding . . . . .	4
1.4 Using multiple parts . . . . .	4
1.5 Dynamic objects . . . . .	5
1.6 summary . . . . .	5
1.7 Our approach . . . . .	5
<b>2 Related Work</b>	<b>9</b>
2.1 Multi-Material Fabrication . . . . .	9
2.1.1 Optical materials . . . . .	9
2.1.2 Multi-material stereolithography . . . . .	9
2.1.3 MultiFab . . . . .	11
2.1.4 MetaSilicone . . . . .	11
2.1.5 OpenFab . . . . .	12
2.1.6 Foundry . . . . .	14
2.1.7 Voronoi Pattern . . . . .	15
2.1.8 3D Tiling of Micro-Structures . . . . .	15
2.2 Computational Molding Processes . . . . .	19
2.3 Accelerating the Fabrication Process . . . . .	21
2.3.1 BlowFab . . . . .	21
2.3.2 LaserStacker . . . . .	21
2.3.3 LaserOrigami . . . . .	21
2.4 Embedding Existing Elements in Fabricated Objects . . . . .	23
<b>3 Concept</b>	<b>25</b>
3.1 Walkthrough . . . . .	25
3.1.1 Step 1: Design and material annotations . . . . .	25
3.1.2 Step 2: Computational design and fabrication of multi-stage mold . . . . .	26
3.1.3 Step 3: Mold assembly, casting, and curing . . . . .	26
3.1.4 Step 4: Removing the mold . . . . .	26
3.2 Multi-Stage Mold Buildup . . . . .	28
3.3 System Overview . . . . .	29

3.3.1	Processing 3D Models . . . . .	29
3.3.2	Casting High-Fidelity Surface Regions . . . . .	29
3.3.3	Hollow and Multi-Material Volumes Inside Objects . . . . .	29
3.3.4	Laser Cutting and Labeling Parts . . . . .	30
3.3.5	Step-By-Step Instruction Manual . . . . .	30
<b>4</b>	<b>Software Implementation</b>	<b>33</b>
4.1	Multi-Material Slicing . . . . .	33
4.2	Venting Pipes . . . . .	33
4.3	Generating 3D printed parts . . . . .	34
4.4	Computing viable cast sequences . . . . .	34
4.5	Validity analysis and pruning of cast sequences . . . . .	35
4.6	Component Integration . . . . .	38
<b>5</b>	<b>Example Designs</b>	<b>43</b>
<b>6</b>	<b>Limitations and Future Work</b>	<b>45</b>
6.1	summary . . . . .	46
<b>7</b>	<b>Conclusion</b>	<b>51</b>

# List of Figures

1.1	Medley [6] created by Chen et al. presents a design tool to embed household items into 3D printable models. Embedding a wire to add flexible capabilities (a); a screw to reinforce a hook (b); a sander made with sanding paper (c); using wax to alter the object even after it was printed (d). . . . .	2
1.2	MetaSilicone [41] presents a technique and tool to imitate harder and softer silicone by infusing the silicone with water or liquid silicone. Infusing (1), a strength test (2), a kidney with harder and softer parts (3), a bunny with harder and softer parts (4). . . . .	2
1.3	Sreetharan [32] designed a fabrication method to create millimeter-scale machines layer by layer. A small fly in the picture. After all layers are glued down on top of each other and lasercutted (e), small hinges can be used to pop out a millimeter-scale device (f). The triangle in the middle of the object is a piezoelectric plate that vibrates when an alternating current is applied. . . .	3
1.4	StackMold's fabrication workflow for making a functional headset in multiple materials: (a) annotating material properties, (b) a sliced version of a multi-stage mold is computationally designed, (c-d) the assembles the mold and casts material according to the instruction manual, (e) the functional headset. . . . .	7
2.1	Willis et al. [39] created Printed Optics. The tubes guide the light emitted from a pocket projector to the eyes of the little figure. When an animation is now played on the projector, the eyes move. . . . .	10
2.2	Willis et al. [39] created Printed Optics. On the left you see the example of a button. The tube behaves as an IR light based contact and at the same time as a small spring to keep the button in place. On the right is an example of a scroll wheel where some parts of the wheel let IR light pass through and other parts do not. . . . .	10
2.3	Examples of the MultiFab platform created by Sitthi-Amorn et al. [31], capable of creating a wide variety of multi material objects with high precision. Objects can even have light transforming effects like caustics (d), Fiber optics (b) or directional light (g). . . . .	12
2.4	MetaSilicones [41] generates a pattern of droplets to inject into a silicone object to achieve the strength requirements set by the user(dotted line). The first example is the stretchability without droplets, second with all droplets at their maximum size and the third is the calculated pattern of droplets. . . .	13

2.5	Foundry [35] gives users the ability to define a material by placing predefined blocks in a pipeline. The result of adding or removing blocks can be viewed in a preview window. . . . .	14
2.6	Bickel et al. [4] used this setup to measure the deformation of the material used in an object. Multiple cameras (Blue) are used to observe the deformation applied by the robot arm (green). The amount of pressure is measured with a force sensor (red). . . . .	16
2.7	The system of Martinez et al. [16] allows for the generation of foam with a wide flexibility range. On the right of the image an example object is shown that is annotated with material stiffness requirements and the generated foam pattern, ready to be 3D printed, to support the object and flexibility requirements. . . . .	17
2.8	Schumacher et al. [29] presented a method to create objects that has flexible capabilities out of a non flexible material. On the left the user annotates how flexible the objects needs to be (red is stiff, blue is soft). In the middle the generated object and on the right the printed object with flexible properties. . . . .	17
2.9	With Metamaterial Mechanisms, Ion et al. [14] created a system to design and generate 3D objects with a mechanical function out of one material without the need for assembly. A and B show how a door handle can work. The software, shown in C, allows the user to use different types of cells to create these mechanical objects. . . . .	18
2.10	MetaMolds by Alderighi et al. [1]. (a) The pressure on the model when the mold is removed in a cetrain direction and the split into two parts. (b) A render of the silicone mold(green) and the 3D printable mold to create the silicone mold (red). (c) End result with the silicone molds in the middle used to cast the figurine and the 3D printed molds to cast the silicone ones on the sides. . . . .	20
2.11	BlowFab [40] from Yamaoka et al. is a fast prototyping technique to create 2.5-dimensional objects by laser-cutting a 2D shape and then inflating it. When removing part of the heat resistant Kapton Film an uneven texture can be created. In the middle, the same 2D-shape is inflated with different cutout in this Kapton Film. The output of the software in the bottom image. . . . .	22
2.12	A headset midway through casting, illustrating StackMold features. . . . .	24
2.13	A simple illustration of the molding process. (a) A figurine to be cast in two materials. (b) The first casting step: props (gray) prevent the blue casting material from leaking into the compartment for the green material. (c) The user separates the mold and removes the props for the green compartment. (d) The second casting step: the user adds the mold for the head region, with the blue head compartment blocked by props. (e) The user separates the mold and removes the props for the blue head compartment. (f) The final casting step; after curing, the user breaks away the entire mold. . . . .	24



3.1	StackMold allows for seamlessly embedding existing components in casted objects, such as this audio speaker. . . . .	25
3.2	(a) The multi-stage mold is build by adding and removing mold parts. (b) Electronic components are fixated in the mold. . . . .	26
3.3	A generated step-by-step instruction manual guides the user through the assembling and casting process. . . . .	27
3.4	Removing the mold after all casting operations reveals the functional audio headset . . . . .	27
3.5	(a) Overhanging structure supported by mold layers. (b) Internal overhanging structure supported by dissolvable material. . . . .	30
3.6	(a) Laser cut features have identifiers engraved. (b) Identifiers of very small parts are looked up in a “small part catalog” . . . . .	31
4.1	StackMold suggests reducing the slicing thickness when parts of the volume are disconnected. . . . .	34
4.2	StackMold generates 3D-printed mold pieces by (a) copying user-specified faces from the original 3D model; (b) extruding to form an initial mold and re-sampling the slices to the region; (c) the 3D mold is further extruded to intersect all mold slices; (d-e) subtracting the slices from the 3D mold to form a “stairstep” pattern. . . . .	35
4.3	(a) A simplified representation of four layers to be cast in two materials, A and B. (b) A <i>volume dependency table</i> for the example. (c) The <i>viable cast sequence tree</i> for the example, showing different possible sequences for casting. . . . .	36
4.4	Properties stored by StackMold for each casting volume, using B2 as the example material volume: (a) material volumes in the same layer which touch B2; (b) volumes touching from above; and if the material volume touches the mold itself (c) above or (d) below. . . . .	37
4.5	Embedding an electronic component into the cast object, using an LED as an example. (a) The user positioned the component on the to-be-molded object. StackMold then (b) places its parametric fixture and (c) sizes it to fit within the intersecting mold layers, (d) adjusts the intersecting layers to prevent interference with the component and fixture, and (e) generates the final fixture. . . . .	38
5.1	Example objects fabricated with StackMold: (a) a silicone turtle character with different colors, (b) a multi-colored silicone figurine with regions in high-fidelity, (c) a vase made of plaster using dissolvable material, (d) a functional audio headset consisting of resin and silicone. . . . .	44
6.1	Some limitations of StackMold: objects with many material changes such as gradients (a) or patterns (b) will require a large number of casting steps. Objects like the tree in (c) require more steps in order to remove temporary props. . . . .	46

6.2	Different slice orientations could be future work. The head of this figurine is sliced in a vertical direction while the rest of the body is sliced horizontally which result in a very detailed head when viewed from the front and a detailed body when viewed from the top. . . . .	47
6.3	A head sliced with dynamic layer thicknesses. Makes layers thinner in tight corners to preserve surface detail. Idea for future work. . . . .	48
6.4	With the motor in this position, there is no possibility for the red striped region to be casted. . . . .	49
6.5	A turtle model with an mdf shell and silicon legs and head (a), same model turned upside down (b). The clamps ensure that the silicone stays attached to the mdf. The head is attached in the same way however this is not visible from the outside. . . . .	49

# 1 Introduction

The process of designing and prototyping small objects has known huge improvements in the last couple of decades with more affordable 3D printers, easy accessible maker spaces and a steady growth in the amount of materials that can be 3D printed, casted or laser cut. However when it comes to the design and prototyping of objects that require multiple materials, there are still a lot of challenges. There are two main options to fabricate a custom multi-material objects today. Number one requires the designer to have access to a multi material 3D printer and the second approach would be to fabricate multiple parts and then combine them together.

## 1.1 Importance of multiple material

While it is easy to create an object out of one material, creating an object that exists out of multiple materials is rather difficult due to a couple of reasons. Number one being that the most widespread and arguably most versatile technique, 3D printing, only supports a limited amount of material. The most popular technique to 3D print is FDM printing, which uses material on a spool that after being heated gets fluid and can be dispensed at a set position. Thus requiring each material to have the property to melt when heat is applied. While this technique is very versatile in the geometries that can be printed, it only supports a limited amount of materials. The usage of multiple materials in one object allows functionalities that are not capable with other techniques.

One example of the usage of multiple materials is Medley [6], a tool to design 3D printable objects with an extensive library of everyday household items and materials to enhance the capabilities of the object. The tool lets a novice embed household items to the model and creates an altered 3D printable object that allows the user to attach the household items and materials to or in it. An example of such a device can be a sander that has a piece of sandpaper attached to it or a reinforced pegboard hook that embeds a metal screw for extra strength. Figure 1.1 showcases some example designs with their real life counterparts.

In the medical industry silicone is often used to imitate skin and organs. But emulating stiffer or harder parts, of an organ for example, in silicone is rather difficult. With MetaSilicone Zehnder et al. [41] solve this by injecting fluids into the silicone. These fluids are water or liquid silicones of various types. Variable levels of flexibility can be achieved by infusing these different fluids into the material. The locations of injecting fluids is calculated in software and the injection itself is done by a modified 3D printer. Figure 1.2

By using multiple different materials it is possible to create functionalities that are often not possible with the use of only one material. For instance Sreetharan [32] designed a fabrication method to create millimeter-scale machines with which they fabricated piezoelectric parts that vibrates when an alternating current is applied to it on the frequency of the AC. Each

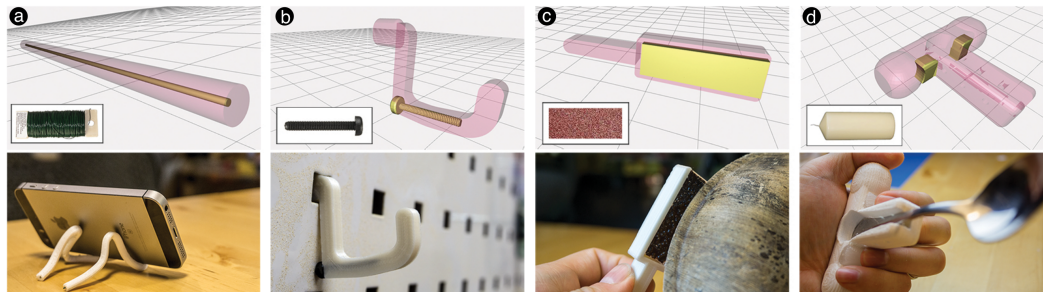


FIGURE 1.1: Medley [6] created by Chen et al. presents a design tool to embed household items into 3D printable models. Embedding a wire to add flexible capabilities (a); a screw to reinforce a hook (b); a sander made with sanding paper (c); using wax to alter the object even after it was printed (d).

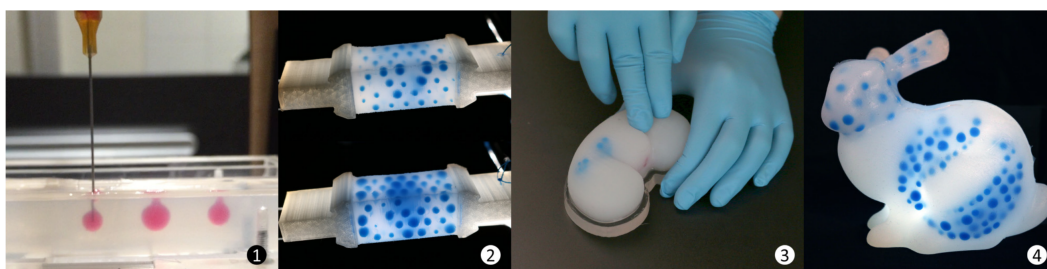


FIGURE 1.2: MetaSilicone [41] presents a technique and tool to imitate harder and softer silicone by infusing the silicone with water or liquid silicone. Infusing (1), a strength test (2), a kidney with harder and softer parts (3), a bunny with harder and softer parts (4).

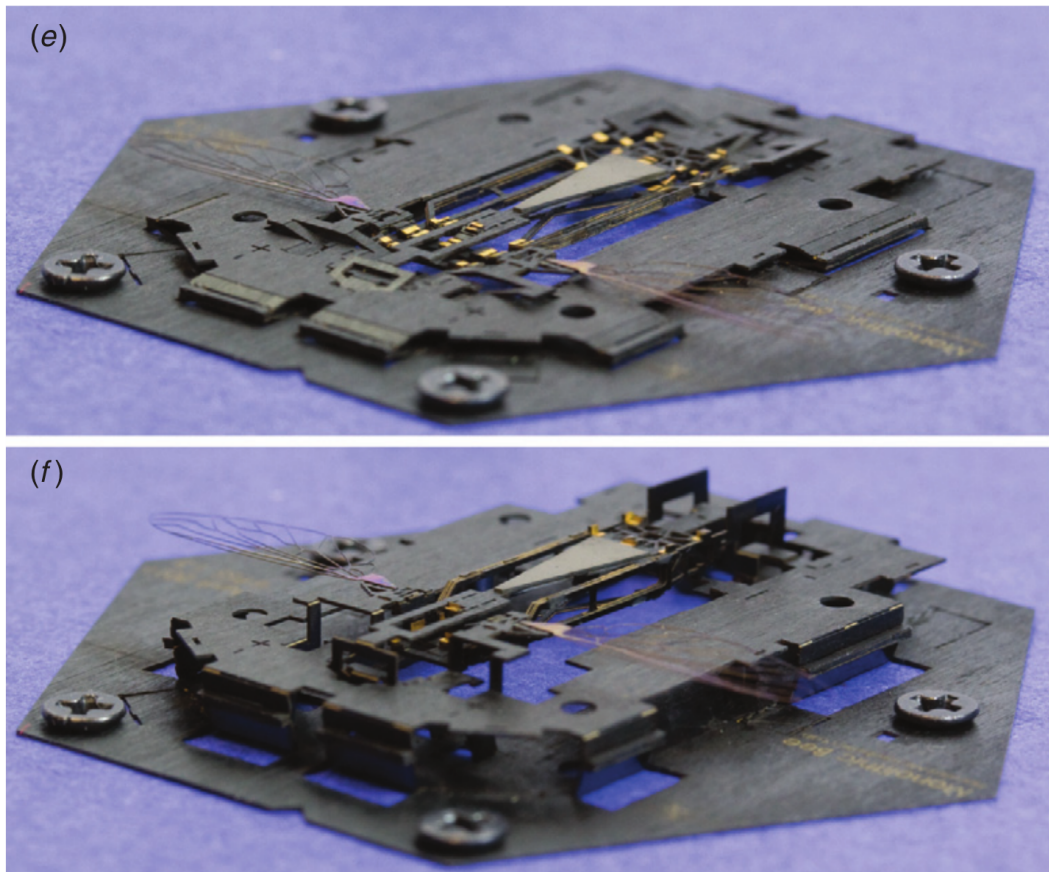


FIGURE 1.3: Sreetharan [32] designed a fabrication method to create millimeter-scale machines layer by layer. A small fly in the picture. After all layers are glued down on top of each other and lasercutted (e), small hinges can be used to pop out a millimeter-scale device (f). The triangle in the middle of the object is a piezoelectric plate that vibrates when an alternating current is applied.

millimeter-scale machine is constructed by laying down one sheet of material, make the correct incisions with the lasercutter and repeat until all material layers are placed down. An adhesive applied to each layer holds all layers together. Hinges and mechanisms, created with the layers and correct incisions, make it possible to pop up a 3D millimeter-scale machine out of the 2D composite sheet material. The piezoelectric part is created by combining a Lead sheet with a Brass sheet. A small fly can be seen in Figure 1.3.

These examples show that there is a big need to create multi-material objects or emulate the behavior of them. It also brings forward the challenge of supporting a wider array of materials to prototype with.

## 1.2 3D multi-material printer

There are multiple advantages of using a 3D printer to create a prototype. First of all, the ability to add a high level of detail. Secondly any possible shape you want is possible, holes, bridges, ... It has to be noted that this is only true if support material is used. Additionally,

3D printing processes are fully automatic and thus do not require active presence of a person. Finally, the designer has no need to have a vast knowledge of the material, machine or process he uses. Most models can be printed rather easily with one push of a button, when the model is already designed. The main disadvantages of a 3D printer are the following: a high setup cost, relatively long fabrication time and a limited selection of printable materials.

The setup cost of a machine that can print multiple materials can be quite high. While more simplistic 3D printers (that only print a single material) are affordable for even the average consumer, machines that are more advanced can quickly rise in price. This can however be circumvented by making use of a makerspace nearby. However, in a makerspace there will be a cue of objects thus not ideal when you are fast prototyping. It also highlights another challenge in creating a prototyping process. It has to be affordable.

A model fabricated by a 3D printer takes multiple hours to be created, even on a low precision of the model. As a result, a full prototyping iteration can take up a lot of time when the designer has to do multiple redesigns. This in will make it impossible to do multiple iterations over the design in one day. Also during prototyping this precision is not always necessary. Makers often only desire sub-millimeter precision in some parts of the object and only in some prototyping iterations. Beyer et al. used this in their Platener paper [3] where they created a tool that automatically generated parts that could be lasercut out of a 3D model, at the expense of detail, to reduce the fabrication time of the object.

As already discussed only a few materials are printable. But also the combination of materials might introduce problems. Soft metals and multiple polymers are possible to be printed however the difference in heat used by the metals and the polymers can make for the polymers to meld. Therefore planning of when to use which material and where to use it is also a challenge.

## 1.3 Molding

Some research has been done around molding custom objects. MetaMold [1] is a software tool that generates the most ideal mold for an object. This increases the amount of usable materials significantly but comes at the cost of a slower fabrication time compared to a 3D printer because the mold also needs to be printed. In a prototyping scenario this is not ideal. The faster a prototype can be created the better. Thus another challenge is speed. Creating a faster prototype is better.

## 1.4 Using multiple parts

A maker could also assemble a model from parts. He could, for example, 3D print one part, laser cut another, cast a piece and use a welder and saw to create the other items. Then the maker needs to assemble all of those together and tighten them up. The strengths of this approach is that it might be faster than 3D printing. However, this process assumes that the designer knows how to weld, cut, assemble, ... all of the pieces and has designed the parts in

a way that they all fit. This is more an expert approach. Prototyping should be easy to do for a novice creator.

## 1.5 Dynamic objects

Most methods described above are great for creating static objects. Objects that do not provide any functionality besides staying the same shape. But when an object needs to be interactive and incorporate electronics or mechanical components, the easy prototype pipeline breaks quickly down. In those cases the maker needs to think about how to fit all components, how and when to embed them, where to fit the wires and allow them room of movement if necessary. This requires time, effort and expertise from the maker and becomes quickly very overwhelming for a novice. A challenge in this field is thus how to embed electric and mechanical parts.

## 1.6 summary

In summary the challenges in this topic are:

1. Support for a broader range of materials.
2. Being able to embed electronics and mechanical parts into the object without the need for the creator to adjust the model by him or herself.
3. A higher fabrication speed. The faster a prototype can be created, the more iterations you can do over them.
4. Prototyping should be more easier for a novice user. Less requirements to start, less experience needed and a more affordable process.
5. Planning on when to use what material and when to add it. Molten metals might melt polymers when they are added in the wrong order.

## 1.7 Our approach

It is comparable with Shape Deposition Manufacturing [18]. In this manufacturing process they use a mill to mill away large pieces out of multiple metal plates. Then they lay each plate upon each other and cast a model. Our solution uses the same technique but with a laser cutter. First the designer loads a model into the software and annotates what part should be made from what material. Then the program generates a vector file that can be cut out of a sheet of material. Then according to what material needs to be placed where, the sheets are layed upon each other and the model can be casted out of the different materials by simply following a step-by-step plan that is also generated by the software. Then the mold can be broken of, layer by layer, and the model is finished. Included in our approach is the ability to place electronic components on the surface. In that case the layered mold will incorporate

holes and holders for the electronic components to keep them in place while casting. Lastly when high detail is required in some regions, these regions of the mold can be 3D printed, thus creating high detail in these parts of the object.

While digital fabrication tools such as 3D printers and laser cutters have moved from industrial settings into makerspaces and workshops, there are still many objects that are challenging for DIY users to create. One important class of such objects is those involving multiple materials with varying physical properties or colors. While high-end industrial printer lines, such as the Stratasys Objet or HP Jet Fusion, can vary some material properties in the same object, they are not capable of producing an arbitrary number of material properties and colors, and their high cost and large size keep them out of reach for most non-industrial users.

As an alternative to these commercial approaches, we draw inspiration from *molding*, a process that has been used to create artifacts throughout human history. The basic approach is to form a mold, which is the negative of the desired object; to cast liquid materials in the mold, such as silicones, resins, or plaster; solidify the material via heat, air, or chemical reaction; and remove the mold, leaving the final object. Thus far, the digital fabrication of mold negatives has relied on 3D printing [1, 15], which can be slow, and restricts the end result to a single material and color. Our approach, StackMolds, produces mold negatives by laser-cutting a sliced version of the mold (Figure 1.4c-d). This not only speeds up fabrication time compared to 3D printing mold negatives, but also enables the use of multiple materials and colors via a multi-stage mold buildup, and allows embedding of external components, such as electronics into the mold material. Similar to other fabrication approaches that allow multiple fidelity/speed tradeoffs within the same model [3, 20], StackMold also leverages the insight that prototypes do not need the same level of precision in every region of the object. Therefore, StackMold supports embedding 3D printed high-detail mold negative regions within the rougher laser-cut mold. Our rapid prototyping process facilitates the fabrication of functional objects for which the look and feel as well as the workings can be tested and fine-tuned in further design iterations (Figure 1.4e).

The contributions of StackMold are as follows:

- a novel DIY multi-stage molding technique to fabricate multi-color, multi-material objects with embeddings such as electronics;
- an algorithm for generating multi-stage molds with selective levels of surface fidelity;
- an end-to-end software environment to enable end-users to easily design objects for StackMold.



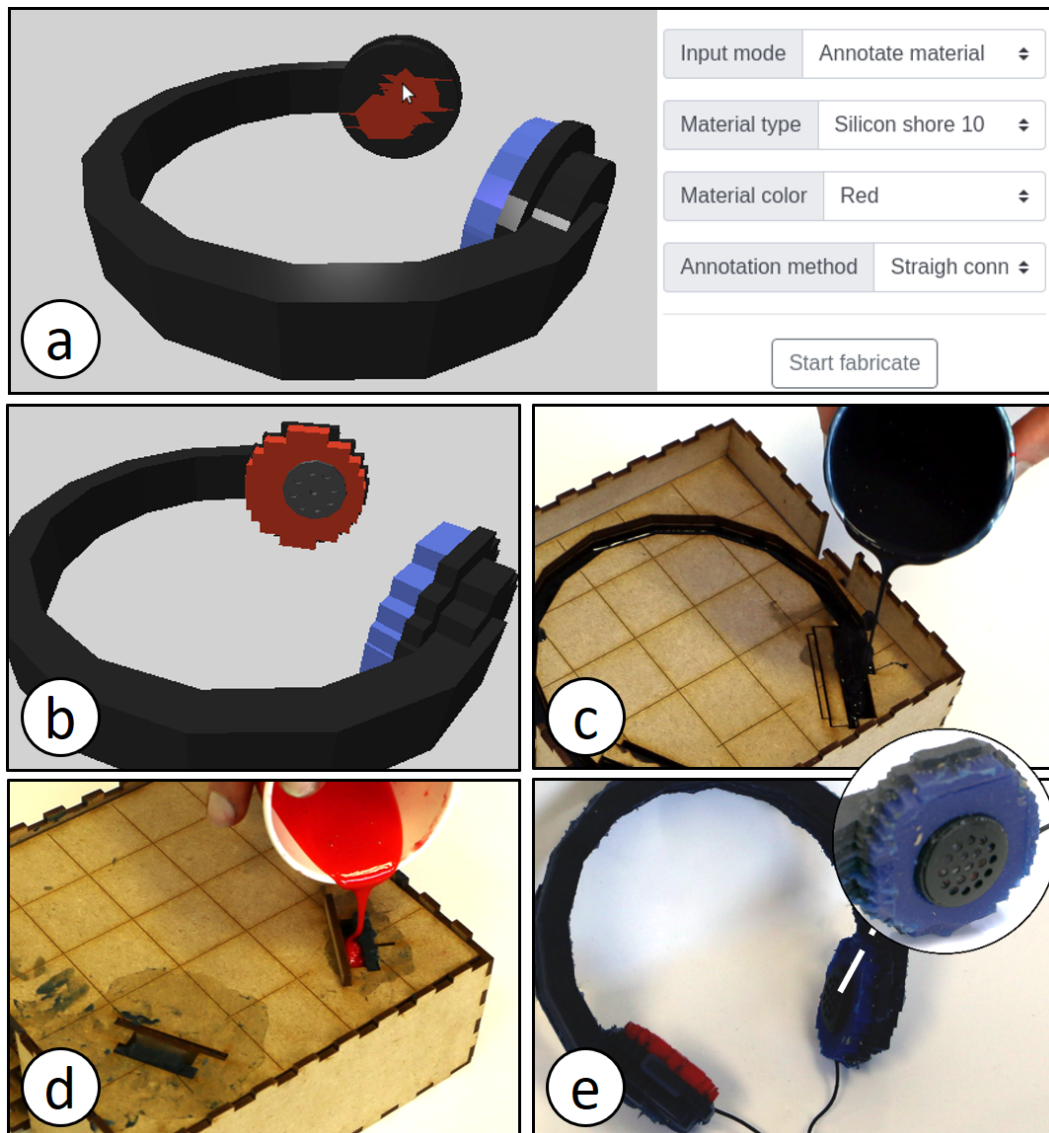


FIGURE 1.4: StackMold's fabrication workflow for making a functional headset in multiple materials: (a) annotating material properties, (b) a sliced version of a multi-stage mold is computationally designed, (c-d) the assembles the mold and casts material according to the instruction manual, (e) the functional headset.



## 2 Related Work

StackMold draws from and builds upon prior work on multi-material fabrication, molding processes, techniques to accelerate fabrication processes, and fabrication techniques for embedding existing elements.

### 2.1 Multi-Material Fabrication

Multi-material fabrication is a long-standing challenge in additive manufacturing. Decades of research and development in this area has resulted in 3D printers such as the Stratasys Connex<sup>1</sup> that support compliant or optical materials [39].

#### 2.1.1 Optical materials

Willis et al. [39] used an extrusion 3D printer with transparent material, Plexiglass<sup>TM</sup>like, to create objects with small embedded tubes that could guide a light source.

Each of these tubes behaves as a fibre optics cable, and could be used to create a small three dimensional display or a digital input device like a button, scroll wheel, slider and touch surface. To create a display the tubes are used to guide the light emitted by a normal display or projector to the correct position as can be seen in figure 2.1.

To create a button, a thin tube is created that is pointed directly at a light detector. Above the tube a button is placed. When a user presses the button, the button bends down the tube and light is no longer directed at the detector. When the user releases tension on the button the tube will act a small spring to push the button back up into position. This is illustrated in figure 2.2.

Similarly, research on powder-based 3D printing with different color bindings [30] has resulted in products that produce plaster models in full-color<sup>2</sup>. Researchers have also investigated stereolithography setups with rotating carousels of resin vats [8] to produce multi-material objects.

#### 2.1.2 Multi-material stereolithography

This is done by using a normal stereolithography 3D printer(A printer that uses light or in this case a laser to transform a resin into a hard polymer) and attaching four rotating resin vats who each have a different material.

---

<sup>1</sup><http://stratasys.com>

<sup>2</sup>e.g., <https://www.3dsystems.com>

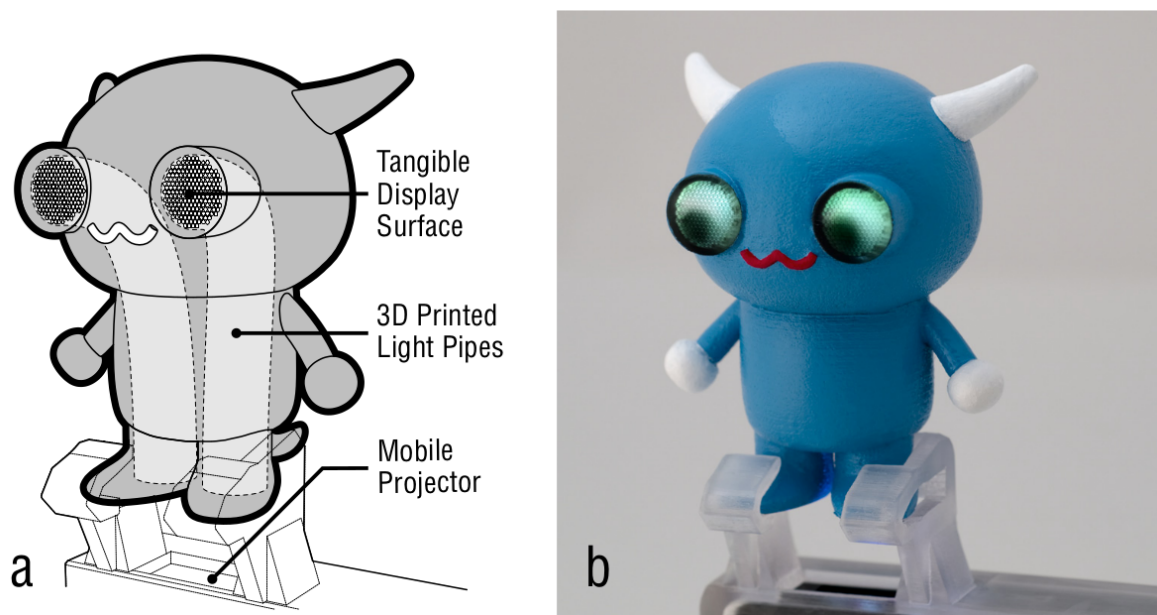


FIGURE 2.1: Willis et al. [39] created Printed Optics. The tubes guide the light emitted from a pocket projector to the eyes of the little figure. When an animation is now played on the projector, the eyes move.

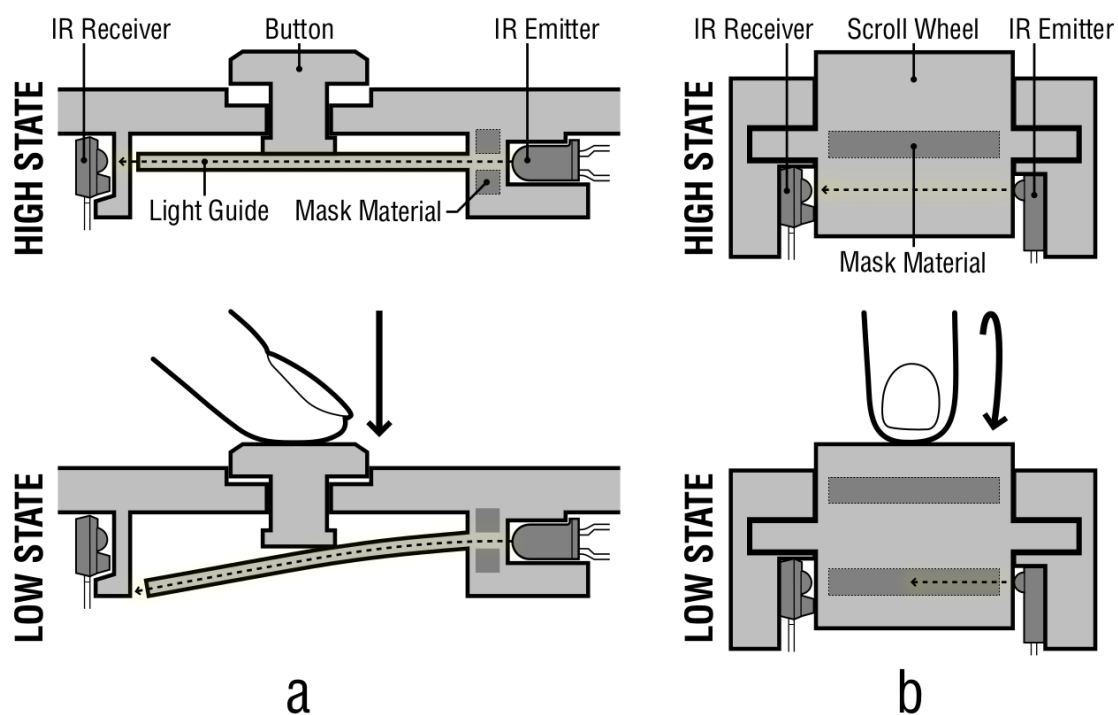


FIGURE 2.2: Willis et al. [39] created Printed Optics. On the left you see the example of a button. The tube behaves as an IR light based contact and at the same time as a small spring to keep the button in place. On the right is an example of a scroll wheel where some parts of the wheel let IR light pass through and other parts do not.

Finally, the Voxel8 FDM printer prints conductive material alongside plastic to enable objects with embedded electronics.

The literature contains a number of special-purpose printers meant for multi-material use. xPrint [37] is a modular machine consisting of syringes, UV lamps, and mechanical stirring functionalities to deposit polymers and living microorganisms. This is a big improvement because filament-, resin-, or powder-based-3D-printed materials are still far away from the level of quality offered by natural or synthetic polymers. With xPrint Wang et al. [37] created an affordable and easy to make 3D printing solution to do create objects made from a wide variety of materials that are not only limited to resins, filaments or powders. Silicons could be used to create strong wearable devices or hydrogels to help with drug delivery.

In a similar vein, MultiFab [31] aims to reduce the cost of multi-material fabrication machinery by complementing low-cost hardware with computer vision feedback; however, the machine's hardware still adds up to around 7000 USD and requires end-users to build their own machines. MetaSilicone [41] takes a different approach and computationally injects liquid dopant droplets into silicone to realize multi-material objects.

### 2.1.3 MultiFab

Sitthi-Amorn et al. [31] created a platform that is capable of creating high precision objects from multiple different materials, adding texture to those materials and it is capable in creating light transforming effects. Examples of this are lenses for LED's, fiber optics or privacy screens which can be found in figure 2.3.

Further more, the platform also provides the user with a 3D scanning capabilities. With the 3D scanner objects could be scanned that in turn could be used in the prototype the user wants to make. An example of this is a circuit board or an electronic component, that is not able to be fabricated with the platform itself. The technique for 3D scanning, that is used in the machine, has a resolution lower than 1  $\mu\text{m}$  and does not require the object to have any form or texture. This allows the platform to scan object made entirely of glass, glossy surfaces and secular materials.

The layer thickness of this setup is approximately 13  $\mu\text{m}$  or 77 layers per millimeter. This resolution higher than the average 3D printer and allows for these very high fidelity objects and light transforming capabilities. Obviously this high resolution also has a time cost. Another feature that decimates the speed of this platform is the use of multiple materials. When only one material would be used instead of two, the speed would increase 10x.

### 2.1.4 MetaSilicone

MetaSilicones [41] changes the behavior of silicone by injecting different fluids into the material. Instead of using different materials, Zehnder et al. created a system to emulate different properties in the same material. They figured that by injecting droplets at just the right place in a silicone model, the stretchability and squishiness of the object in that region changes. Their contribution was a software platform and injection machine to create

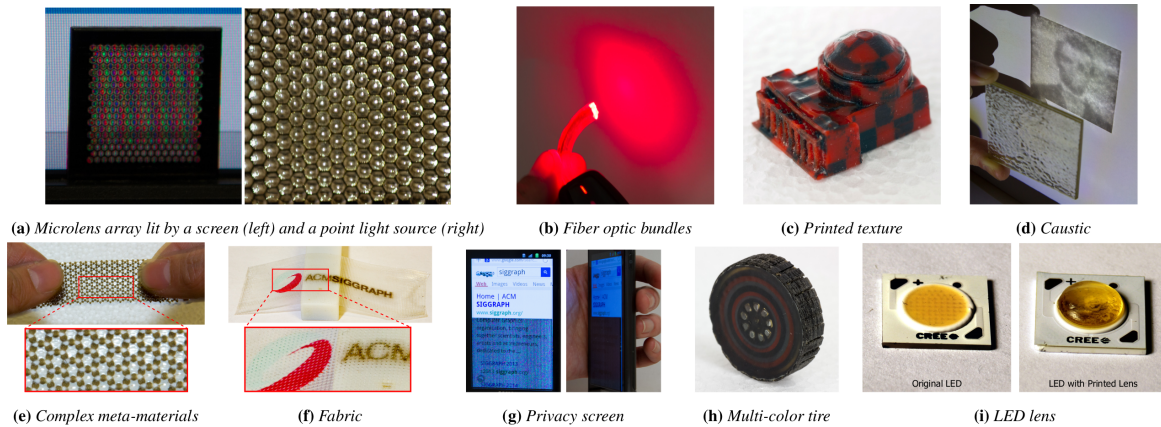


FIGURE 2.3: Examples of the MultiFab platform created by Sitthi-Amorn et al. [31], capable of creating a wide variety of multi material objects with high precision. Objects can even have light transforming effects like caustics (d), Fiber optics (b) or directional light (g).

these objects. The machine was a modified 3D printer with a thin needle that is capable of dispensing the small droplets into the silicone mold.

The software is used to annotate the regions the material needs to be harder or softer and to calculate exactly how many, how thick and where to dispense the droplets. The droplets can not merge with each other because that would create the risk of them preventing further fabrication of the object. Figure 2.4 shows the general idea of the droplet infusion and its effect on the stretchability.

Despite this commercial and research progress, most of these technologies are still in their infancy, or result in high-end products out of reach for makers because of their price and complexity. As such, when making multi-material objects, makers are still largely restricted to dual-head FDM printers or devices that combine a number of filaments into a single strand<sup>3</sup>. Instead of relying on complex machinery, computer systems can also instruct users on how to assemble various materials using a motor controlled laser pointer [12] or by computationally designing attributes to assist the fabrication process of objects [10]. StackMold takes inspiration from these approaches and generates step-by-step instructions to assist users in casting advanced multi-material objects.

To support users in designing and specifying multi-material objects, researchers have investigated scripting [36] and visual programming approaches [35] to facilitate specifying hierarchical material compositions and advanced material gradients. StackMold’s software environment also facilitates assigning materials to 3D volumes using 3D surface brushes similar to by Brochu et al. [5].

### 2.1.5 OpenFab

With OpenFab, Vidimče et al. [36] created a system to define the buildup of an object with a shader like programming language. A material or texture can be described in code and

<sup>3</sup><https://www.mosaicmfg.com>

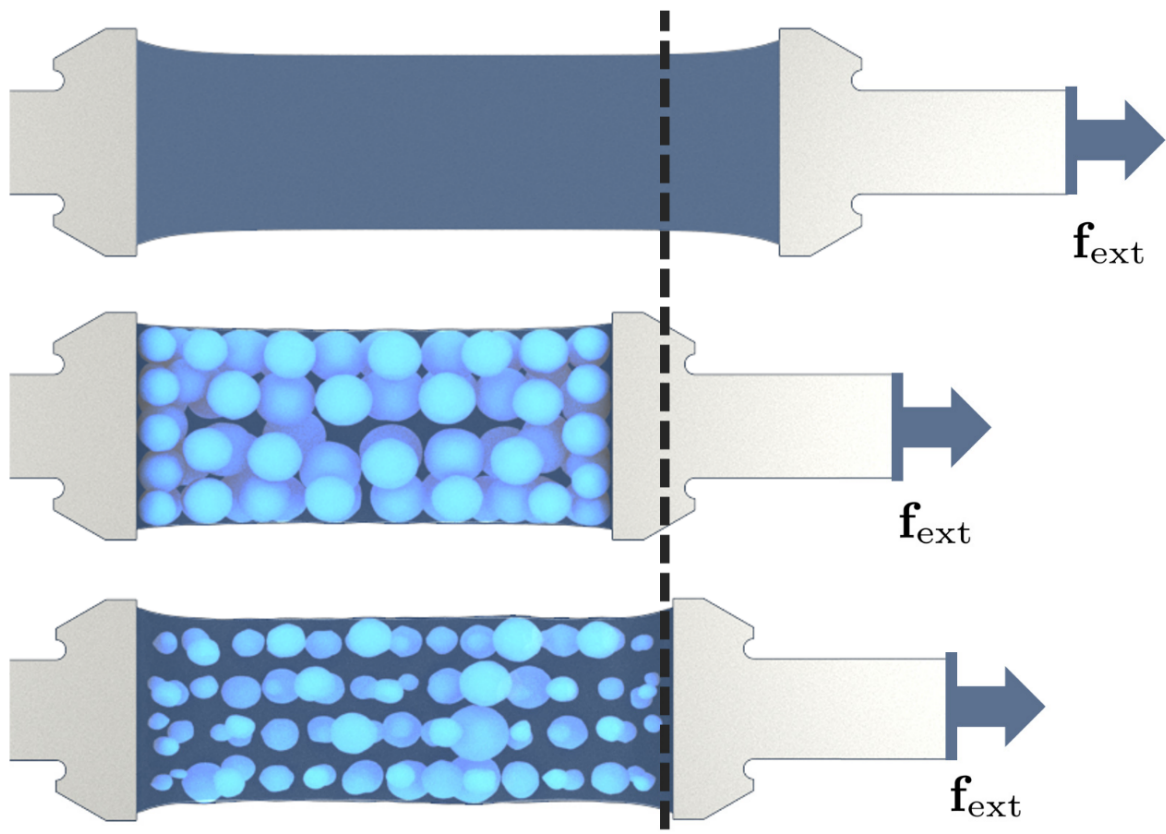


FIGURE 2.4: MetaSilicones [41] generates a pattern of droplets to inject into a silicone object to achieve the strength requirements set by the user(dotted line). The first example is the stretchability without droplets, second with all droplets at their maximum size and the third is the calculated pattern of droplets.



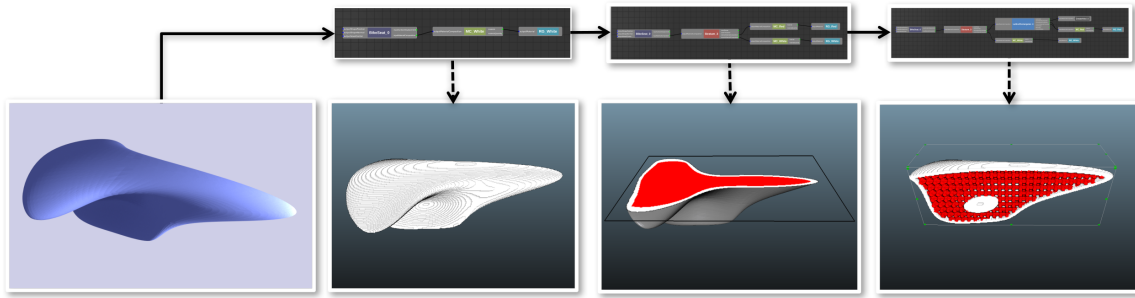


FIGURE 2.5: Foundry [35] gives users the ability to define a material by placing predefined blocks in a pipeline. The result of adding or removing blocks can be viewed in a preview window.

then applied to a geometry. Note that this means that the defined material and geometry are independent from each other. Because the material is defined as a simple code independent from the geometry, it could be reused on to other geometries.

In practice the process of designing the material is done in two steps. Step one is a *tessellation*. This step creates the surface of the object and describes what needs to be done to it. With the help of code, textures and properties can be added to the surface in this step. The next part in the process is *voxelization*. In this step the volume is described of the object. The volume of the geometry is broken down into small voxels (little cubes) and for each of these the volume code is executed.

After these two steps a material is created and can be used, in combination with a 3D model and multi-material 3D printer, to generate complex multimaterial objects.

### 2.1.6 Foundry

Since then Vidimče et al. [35] build further up on OpenFab. One of the mayor draw backs of OpenFab was the programming approach of It. It gives the designer a lot of freedom to do whatever he or she wishes to do with a material or object, but it also expects the user to have a deep knowledge of shader like programming and the new commands and features that come with OpenFab. However, programming is a difficult to learn skill and writing code for a shader can be even harder to learn so it is not really usable for Novice users.

Therefor Vidimče et al. created Foundry. Foundry is build on top of the OpenFab engine and utilizes a connect-the-blocks programming structure to create materials. The user can select multiple different operator blocks and connect them through each other creating a material pipeline. The result of adding or removing a block can be viewed in a preview window of the object where it can be sliced to view the inside. An example of this can be found in Figure 2.5.

Foundry allows for the creation of a very complex material without the need of knowing how to program.

As an alternative to multi-material additive manufacturing, techniques have been proposed to emulate material characteristics by changing the micro-scale structure of objects printed with a single material.



Bickel et al. [4] created a system with which he can measure and observe the deformation of a material. A set up consisting of multiple cameras, a robot arm and a force sensor, can observe and map the requested material deformation behavior (Figure 2.6 shows the setup). This information is then used to create 3D printable micro structures which combined in an object can result in the required material deformation behavior. Examples of these objects are flip flops or cushions of a stool. This method makes it very easy to replicate not only an item his visual form but also the characteristics and functions of the material used of this object.

Schumacher et al. [29] and Martinez et al. [16] optimized respectively the 3D tiling of micro-structures and voronoi pattern layouts, to realize intricate compliant structures.

### 2.1.7 Voronoi Pattern

The application of Martinez et al. [16] allows the user to annotate how rigid or flexible certain parts of a geometry need to be. Then the software calculates a pattern of fibers in the shape of the object that creates a foam like structure. The density of the fibers depends on the requested flexibility in that specific position of the object. The pattern can then be printed by a 3D SLA or SLS printer with only one material. SLA and SLS techniques both allow for printing without support material. This is important because if support material was used inside the foam it would be more than likely that it would never come out again and thus block any ability to compress or stretch the object. The foam is procedurally generated and to optimize the execution time of the foam generation, the voxel calculations are implemented in OpenCL and executed on a graphics card.

This pattern allows for smooth transitions from highly rigid to very fluffy and flexible parts in any shape with no restriction in dimension or form. An example of how the foam looks and is structured can be found in Figure 2.7. In the same figure there is also an example of how a figure with different flexibility's is build up with the foam.

### 2.1.8 3D Tiling of Micro-Structures

While the Voronoi pattern is very flexible to use and does not have any dimensional constrains, it requires a flexible material and the use of an SLA or SLS printer. Although not the most costly, they still cost significantly more than a simple FDM printer. Schumacher et al. [29] presented a tile based approach where each tile can be manufactured with any 3D printer and relatively stiff material. different types of tiles are used. Each type has his own functionalities and capabilities. By thickening or reducing the material thickness of a tile the functionalities can be tuned to the requested abilities. To make use of multiple material functionalities at the same time, different types of tiles are interlaced in a pattern to accommodate the requested properties. The software is comparable to the work of Martinez et al., where the user can define how rigid or flexible the material has to be on a specific position and then the most optimal tiling is calculated. An example of this technique can be found in Figure 2.8.

The downside of this model is that, because a fairly non flexible material is used, no outer skin or shell can be made for the model so from the outside, the tiles are still visible in the model as can be seen in Figure 2.8 on the right. Also compared to the Voronoi pattern, the

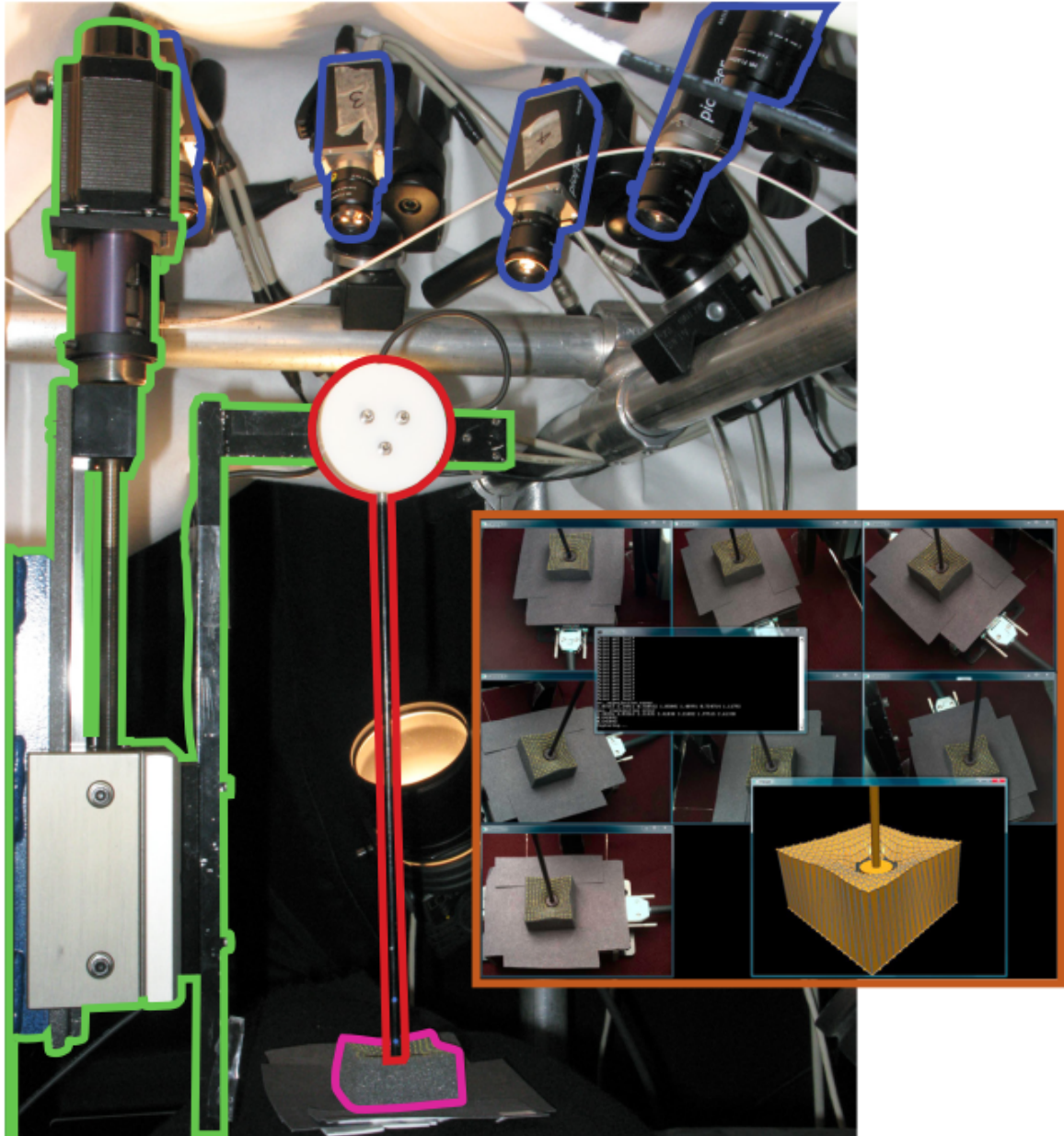


FIGURE 2.6: Bickel et al. [4] used this setup to measure the deformation of the material used in an object. Multiple cameras (Blue) are used to observe the deformation applied by the robot arm (green). The amount of pressure is measured with a force sensor (red).

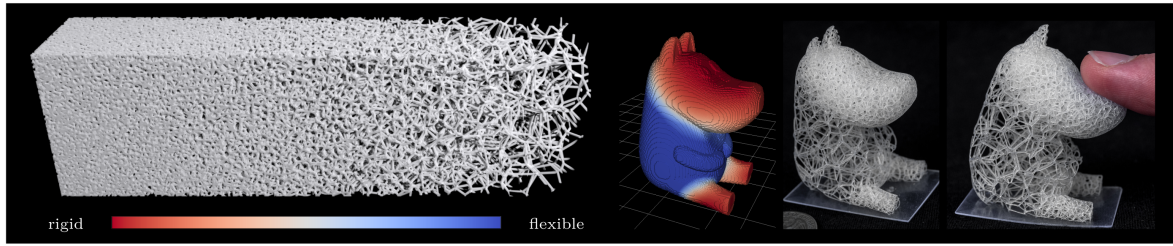


FIGURE 2.7: The system of Martinez et al. [16] allows for the generation of foam with a wide flexibility range. On the right of the image an example object is shown that is annotated with material stiffness requirements and the generated foam pattern, ready to be 3D printed, to support the object and flexibility requirements.



FIGURE 2.8: Schumacher et al. [29] presented a method to create objects that has flexible capabilities out of a non flexible material. On the left the user annotates how flexible the objects needs to be (red is stiff, blue is soft). In the middle the generated object and on the right the printed object with flexible properties.

tiles require some space, so they have a dimensional limit. It is hard to construct a small object with rapidly differing material stiffness levels.

Using these techniques of Bickel et al., Schumacher et al. and Martinez et al., researchers also showed how to fabricate shape-changing structures [23] and functional mechanical objects [14]. The shape-changing structures of Panetta et al. [23] are created with small tiles that each have a different kind of flexibility. The user can then texture a model with the blocks and see how it changes shape. Ion et al. [14] allows the user to experiment with different kinds of 3D printable cell blocks. The user can create a model with these blocks and then set locked points on this model that do not move and set points and working forces on that model from multiple directions. The resulted movement/deformation of the model can then be observed in a simulation. This allows the user to create mechanically working objects that can be 3D printed in one single bendable material. Some examples of these items are: a pair of pliers, a latch and a door handle (door handle can be seen in Figure 2.9).



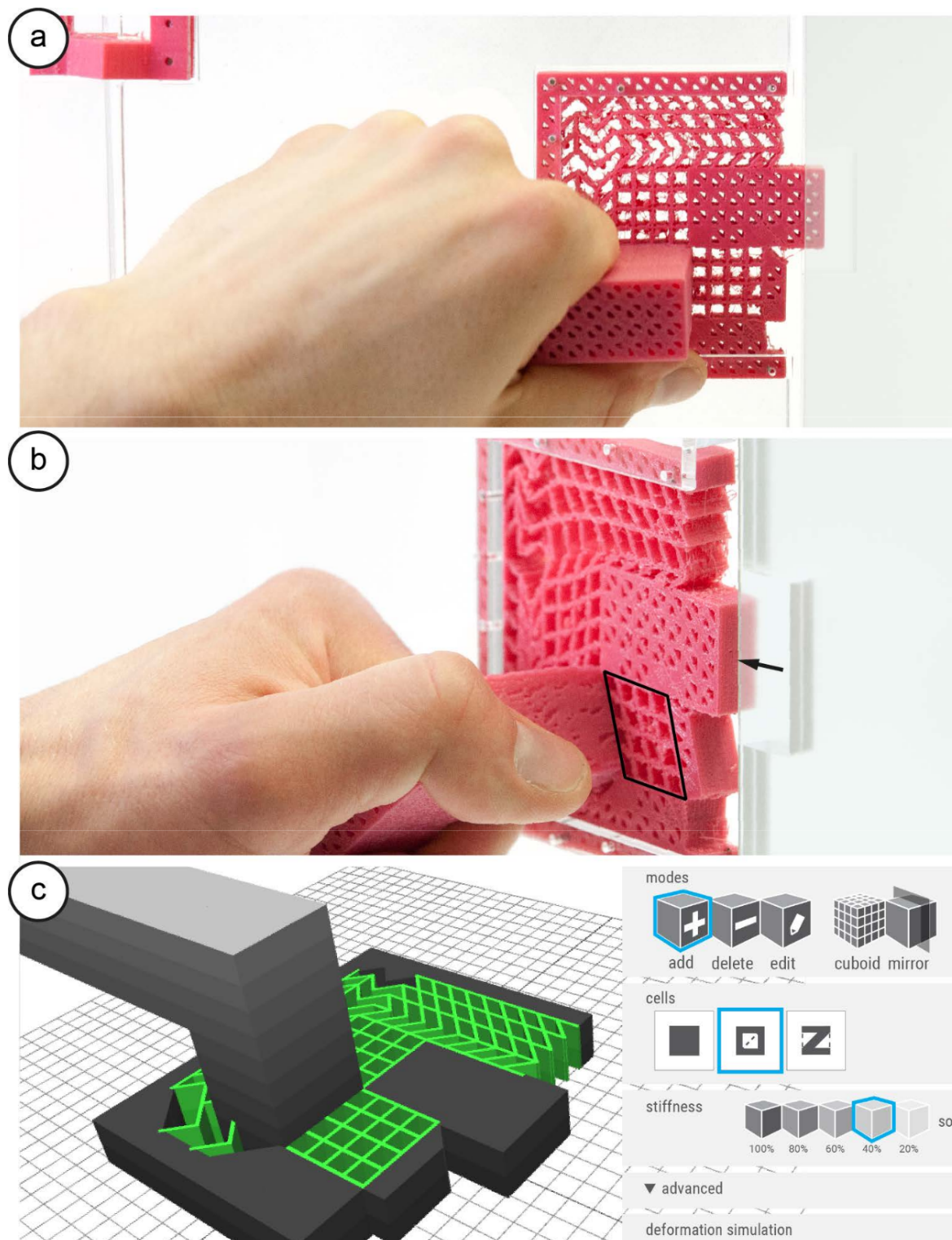


FIGURE 2.9: With Metamaterial Mechanisms, Ion et al. [14] created a system to design and generate 3D objects with a mechanical function out of one material without the need for assembly. A and B show how a door handle can work. The software, shown in C, allows the user to use different types of cells to create these mechanical objects.

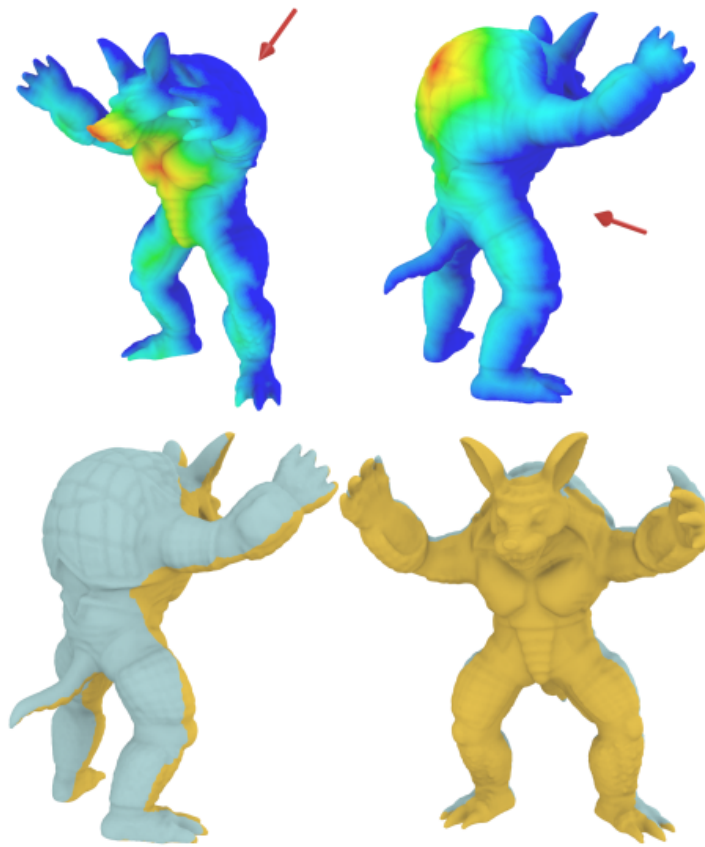
## 2.2 Computational Molding Processes

Molding is an ancient technique for rapidly producing 3D shapes as it allows for casting large quantities of natural materials (e.g. resins, polymers, and plaster) into a precise shape using a mold. Traditionally, a negative shape of an existing object was created to serve as a mold for copying that object. Nowadays, digital fabrication technology allows for fabricating a mold directly from a digital version of the desired object. Herholz et al. [13] investigated how to produce molds using a 3-axis CNC milling machine. As 3-axis milling requires every point on the surface to be reachable by the drill bit (height field constraint), the mold has to be split into parts and some local overhangs required distortion to satisfy the height field constraint. Merz [?] built an early additive manufacturing process to realize molds. As the material deposition technique could only achieve layers thicker than 0.125mm, a CNC milling step was required at every layer to produce more-intricate mold features.

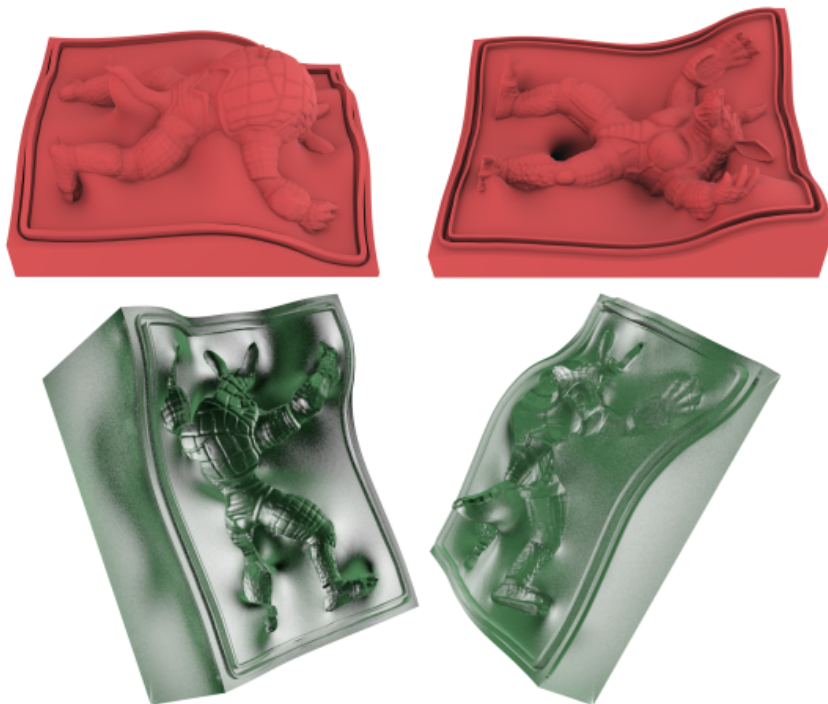
The current generation of additive manufacturing technology allows for fabricating highly intricate molds. As removing mold material in small cavities is often challenging, several research projects focus on fabricating flexible molds that are easy to remove and can be reused for casting multiple copies. A flexmold [15] is a thin flexible shell fabricated with 3D laser-sintering. Flexmolds are computationally designed with sufficient cuts to extract the solidified material without breaking the mold. The code also determines where to place air vents to prevent pockets of air being trapped inside the mold. when the cast is cured, the outer mold can be removed by peeling it of the object. Because its made from a flexible material, the mold can always be removed from the object because it is easily deformable. To keep all the seams together and to make it easier for the user to do so, the model has small hooks are generated at both sides of the seem to allow the user to tie both sides together. Casting needs to happen in a specific angel on the object, therefor a support stance is also printed to hold the mold in the correct rotation and into place.

MetaMold [1], on the other hand, presents a similar technique but first generates a solid 3D printable meta-mold to produce the actual flexible mold through an additional silicone casting step. The resulting mold will exist out of two pieces and is optimized to prevent any form of stress on the casted object when it needs to be removed. To do this Alderighi et al. designed a tool that tests out multiple directions and tries to find the best approach with the least amount of tension on the object. Using silicon for the mold material will also help with reducing stress on the casted model while removing because of it's flexible nature. Figure 2.10 shows the pressure on the model in one orientation and its split. The 3D generated 3D print, the mold for the mold and the actual mold. and finally it also shows the product with the molds.

In contrast to these approaches, StackMolds are fabricated by means of laser cutting. Our multi-stage molding technique is the first approach that allows for precisely casting multiple materials in a single mold. StackMolds are, however, not reusable as they are intended to produce one version of a prototype before transitioning to the next prototyping iteration.



(a)



(b)



## 2.3 Accelerating the Fabrication Process

3D printing has revolutionized the prototyping process enormously. But while the capabilities of 3D printers are rising rapidly, printing with fine detail is still a long process. To speed up this process a number of rapid prototyping techniques have been proposed. One approach is to take two-dimensional materials and make them 3D via deformation. Yamaoka et al.'s BlowFab [40] introduced a technique to create 2.5-dimensional objects by laser-cutting a 2D shape and then inflating it. With LaserStacker, Umapathi et al. use the capabilities of a laser cutter to weld multiple layers of plastic together [33], and Mueller et al. use a laser cutter to apply heating to bend 2D plastic sheets into 3D shapes under the force of gravity [19] in their project LaserOrigami. Other approaches use 2D slices that are fit together to form a 3D object [9, 17, 25]. Muntoni et al. decompose a 3D shape into 2.5D "blocks" that can be milled with a CNC machine and then assembled.

### 2.3.1 BlowFab

They created a three material layered material that exists out of PET, Kapton Film and Masking Tape. The laser cutter cuts out the shape of the object from the object and the user removes the masking tape where he or she wants to meld both sides together. This means that masking tape is not removed at the inflatable area. Both sides are then placed on top of each other and heated to 120 °C. At this temperature the PET layers will meld en become stretchable but the Kapton Film will remain its strength because of their much higher heat resistance of 260 °C. So the multilayered film becomes flexible because the PET went from a rigid state to a stretchable and flexible state but the Kapton Film will only be flexible thus prevent any stretchability. When the object is hot, the user has to blow air in the object through a straw and allow the object to cool down. When the object is cooled down, the PET layer will be hard again and thus the object will be solid and retain its shape.

By making small engravings or cuts in the Kapton Film, the PET is allowed to expand. This can be used to create uneven surfaces, as can be seen in Figure 2.11, or bend the existing shape in some way. To predict this kind of bendable behavior a simulator was created. A small cut could also be made in the PET layer which allows the user to later remove that part of the model, and thus create a hole, that could be useful when creating a vase for example. The method of Yamaoka et al. allows for the very fast development of prototypes, but they do require the design and creative skills of the user to come up with the 2D shape that will eventually become a 3D model when inflated. This technique is also restricted to a small number of objects because of its 2.5 dimensional nature.

### 2.3.2 LaserStacker

\*TODO\*

### 2.3.3 LaserOrigami

\*TODO\*



FIGURE 2.11: BlowFab [40] from Yamaoka et al. is a fast prototyping technique to create 2.5-dimensional objects by laser-cutting a 2D shape and then inflating it. When removing part of the heat resistant Kapton Film an uneven texture can be created. In the middle, the same 2D-shape is inflated with different cutout in this Kapton Film. The output of the software in the bottom image.



Another approach is to quickly add bulk to a 3D-printed object via already-existing material, using the capability of the 3D printer to add detail where needed. Mueller et al. [20] and Beyer et al. [3] used a combination of Lego blocks, 3D printing, and laser cutting to quickly prototype objects, while Chen et al. extended the approach to allow for quick fabrication of large, high-detail 3D-printed objects by filling the volume with plastic building blocks [6].

StackMold takes inspiration from both of these approaches. We use a laser cutter to create slices, allowing us to quickly build up a rough mold which can be rapidly filled with modeling material. We also 3D-print high-detail elements where necessary, enabling selective level of detail for the eventual output.

## **2.4 Embedding Existing Elements in Fabricated Objects**

Embedding existing components can enhance the mechanical and electrical properties of the object. Medley [7] presents a library of everyday household items that can be inserted into 3D printed objects to overcome the limitation of the small amount of materials that can be 3D printed. Their library, and accommodating software, provides extra flexibility to 3D printable objects. Materials that can be included are wires, nuts, sponges, screws, ... A wire could be included into a 3D print to make a flexible part, bendable and help retain its shape, the sponge can be used as a grip for a device or the screw could be used to give a hook more strength. The software tool allows the user to easily integrate the objects in the 3D model and provides instructions on how to craft the model. The most ideal path to insert an object or material into the object is searched and used in the plan to embed the item. If the item is totally submerged inside the object, it is possible for the software to print over the object and thus totally embedding it.

In a similar vein, Enclosed [38] and RetroFab [24], shows how structures can be computationally designed and fabricated to hold electronic components in place. In this context, researchers also explored techniques and supporting tools to embed conductive traces and capacitive pads in objects using dual-head extrusion printers and conductive filaments [26, 28]. Alternatively, 3D printed channels can be filled manually using conductive paint [26], strips of copper [34] or automatically using special-purpose wire winding machinery [2]. Although these research efforts facilitate embedding of circuit traces in objects, seamlessly embedding electronic components in a fabricated object remains challenging.

As an alternative approach, several research projects investigate how to turn fabricated 2D substrates in 3D interactive shapes. FoldIO [22] and PrintGami [11] present techniques for conductive inkjet printing of circuits that are respectively folded in 3D shapes or inserted in 3D printed objects. Going beyond flexible substrates, Silicone Devices [21], presents a fabrication procedure to prototype stretchable circuits with embedded components that can be wrapped around 3D objects to add interactivity.

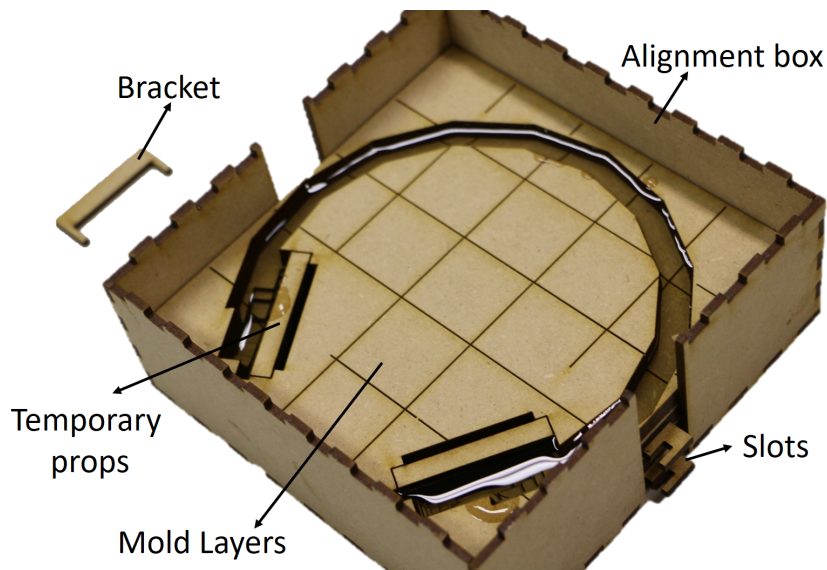


FIGURE 2.12: A headset midway through casting, illustrating StackMold features.

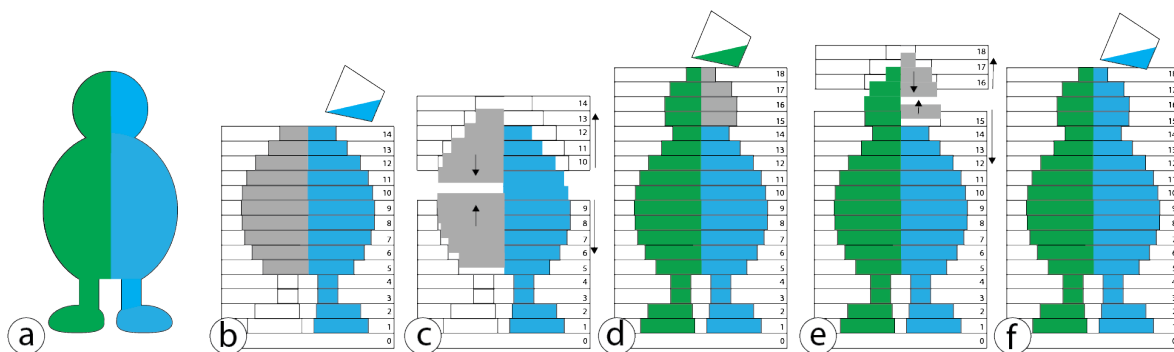


FIGURE 2.13: A simple illustration of the molding process. (a) A figurine to be cast in two materials. (b) The first casting step: props (gray) prevent the blue casting material from leaking into the compartment for the green material. (c) The user separates the mold and removes the props for the green compartment. (d) The second casting step: the user adds the mold for the head region, with the blue head compartment blocked by props. (e) The user separates the mold and removes the props for the blue head compartment. (f) The final casting step; after curing, the user breaks away the entire mold.

## 3 Concept

### 3.1 Walkthrough

In this section, we provide a walkthrough of the StackMold design and fabrication process, highlighting the ability of the user to utilize multiple materials and colors, and to embed external objects. We illustrate the fabrication of a functional audio headset, consisting of soft earpads, a rigid headband, embedded speakers, and several colors.

#### 3.1.1 Step 1: Design and material annotations

The user starts the design process by loading an STL file of a headset model into the StackMold software environment. The material panel shown in Figure 1.4a offers 3D brushes to annotate the mesh in various colors and materials, including plaster, resins, and silicone with varying levels of softness. The user first selects black resin as the default base material, then, as shown in Figure 1.4a, uses the soft silicone brush to specify the material properties of the ear pads, one in blue and one in red. Adjusting the intrusion depth of the 3D brush controls the thickness of the compliant ear pads. Using the embeddings menu panel, shown in Figure 3.1, the designer positions speakers in the earpads.

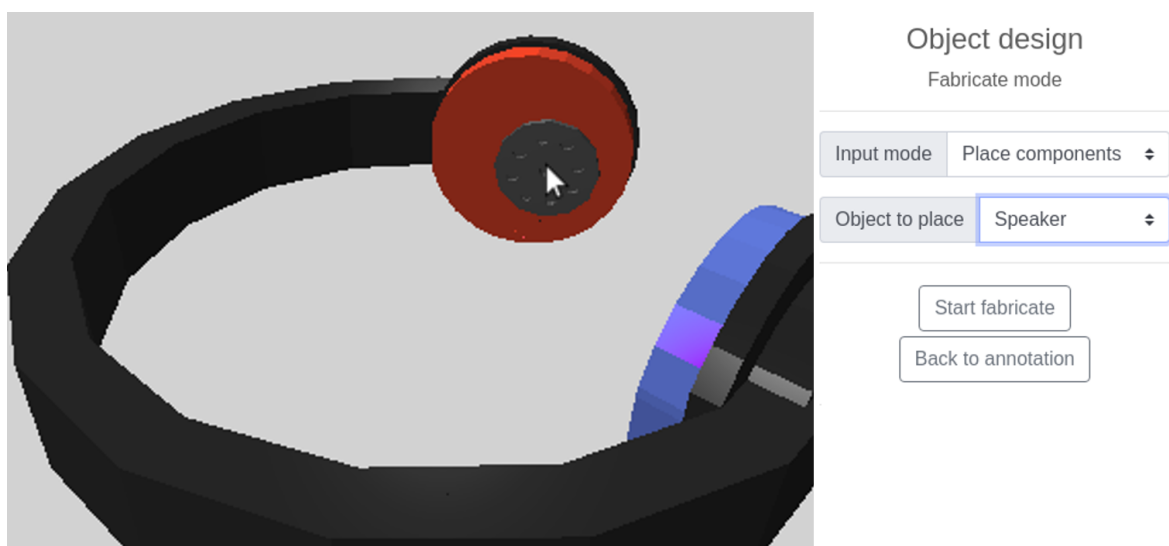


FIGURE 3.1: StackMold allows for seamlessly embedding existing components in casted objects, such as this audio speaker.

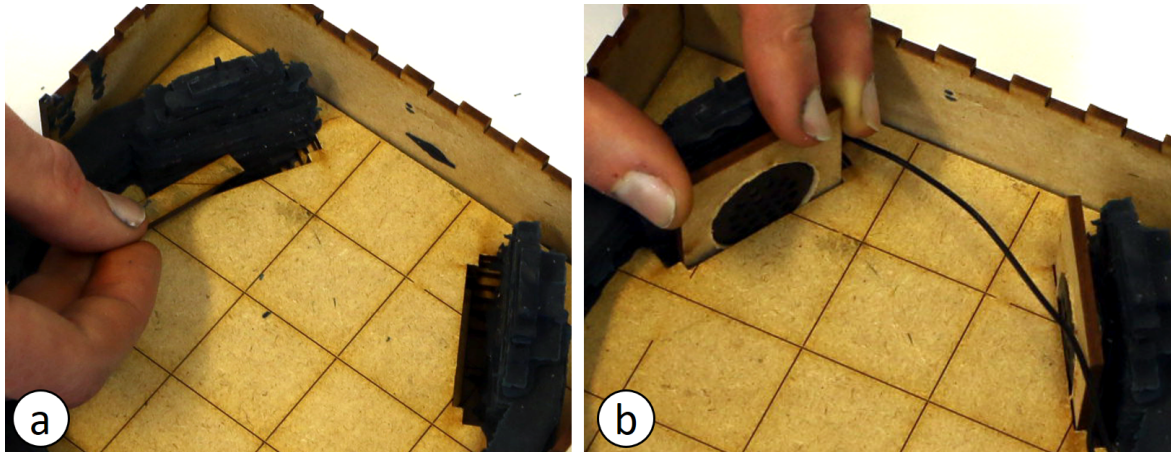


FIGURE 3.2: (a) The multi-stage mold is built by adding and removing mold parts. (b) Electronic components are fixated in the mold.

### 3.1.2 Step 2: Computational design and fabrication of multi-stage mold

StackMold's fabricate button renders a preview of the to-be-fabricated object. The default is to produce a fast-to-fabricate, lower-detail mold via laser-cutting a sliced representation of the mold (Figure 1.4b). StackMold outputs SVG files for laser cutting the mold slices, as well as mounts for external parts. All parts are annotated with embossed labels, referenced in the step-by-step instructions that StackMold generates to walk the user through the mold buildup and casting operations.

### 3.1.3 Step 3: Mold assembly, casting, and curing

Once all mold parts are fabricated, the casting process starts by assembling the mold according to the instructions (Figure 3.3). Depending on the complexity of the mold, this process may involve multiple steps of adding or removing mold parts and casting with and curing different materials. For the headset, StackMold instructs the designer to assemble the first layers of the mold and cast with the black resin base material (Figure 1.4c). After curing the resin, the designer assembles the mold compartments for casting the silicone earpads by removing and inserting mold parts according to the instructions (Figure 3.2a). As shown in Figure 3.2b, StackMold embeds computationally designed fixtures to hold the speaker in place during the casting process. The designer proceeds with the assembly and casting steps, finishing the process after 2 curing steps (red and blue silicone is cured at the same time).

### 3.1.4 Step 4: Removing the mold

Once the casting process is finished, the designer removes the mold, leaving the functional headset behind (Figure 1.4e). Removing the mold is convenient as every layer has break lines engraved, making it easy to snap off pieces (Figure 3.4). Figure 1.4e shows the prototyped functional headset which can be used with an audio cable. Fabricating this functional headset took 120 minutes in total which is faster compared to 3D printing the headset in 6 hours and 5 min with an Ultimaker 3 (low resolution and single material). Furthermore, our prototype

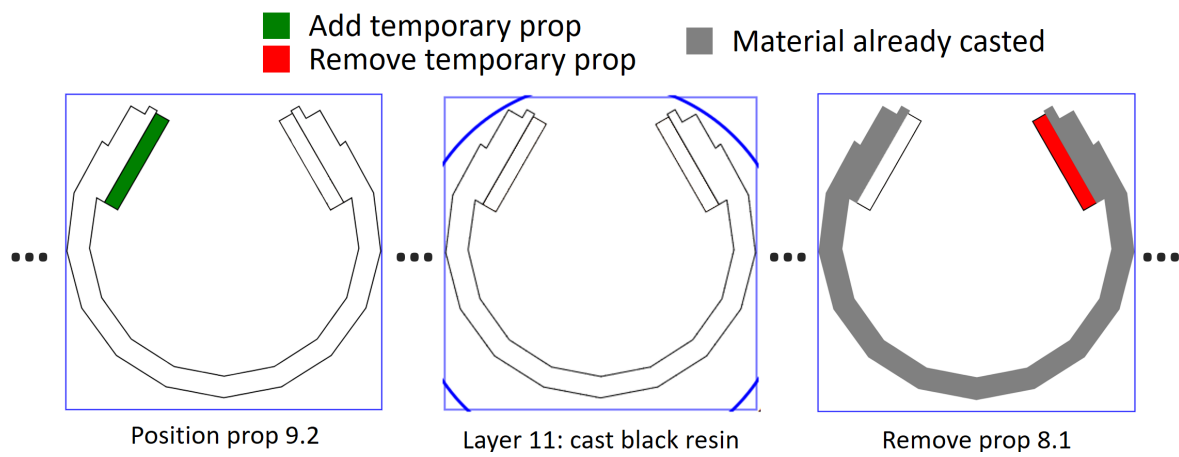


FIGURE 3.3: A generated step-by-step instruction manual guides the user through the assembling and casting process.

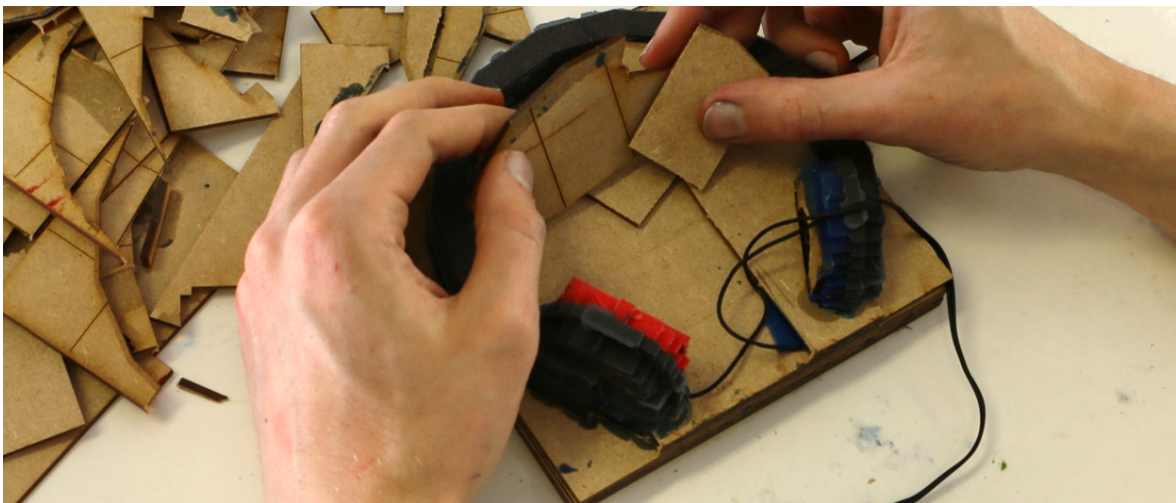


FIGURE 3.4: Removing the mold after all casting operations reveals the functional audio headset

consists of multiple materials, colors, and embedded electronics, which is not possible with conventional FDM 3D printing processes.

## 3.2 Multi-Stage Mold Buildup

StackMolds have a box-like shape on the outside but expose intricate shapes on the inside to cast the desired object (Figure 2.12). To prevent mixing of different materials inside the mold, only one material is cast at a time; this is accomplished by dividing the model into compartments using temporary laser cut props (Figure 2.12). These compartments need to be accessible for casting and temporary props have to be removable. Therefore, StackMold uses a novel multi-stage buildup which involves adding temporary props and mold layers between subsequent casting operations; during these operations, new compartments are configured.

As illustrated in Figure 2.12, a multi-stage mold design consists of several laser-cut features. To prevent between-layer leaks, StackMold adds slots for a laser-cut bracket to squeeze together all layers involved in each casting operation. Additionally, StackMold generates a custom alignment box with finger joints to precisely align all layers while casting. To simplify the removal of the mold after the casting process is completely finished, a grid pattern added to all mold layers, enabling the mold to be broken apart.

Figure 2.13 shows a multi-stage mold build-up for casting a basic figurine in blue and green silicone. In contrast to the feet, which have distinct casting volumes, the two materials are in direct contact in the body and head regions and thus require assembly of compartments to prevent the two colors from mixing. In preparation of the first casting step, the mold is assembled until layer 14 (Figure 2.13b). Temporary laser cut props (gray) are inserted in one half of the body to prevent blue silicone from leaking into the volume reserved for green material. After curing the blue silicone, the props are removed by splitting the mold at layers 9/10 (Figure 2.13c). To facilitate the removal of temporary props, more mold layers on top or below can be detached. Next, the final mold layers and temporary props, to cover the blue compartment of the head, are added. After casting the entire green compartment, the blue compartment of the head is cast (Figure 2.13f). Prototyping this figurine takes three casting steps in total. While it might seem that only one casting step per material should be required, this is not a valid solution as the head would prevent splitting the mold at layers 9/10 to remove the props after the first casting step.

The example discussed in this section highlights the basic principles of our multi-stage mold buildup. The example figurine, however, has smooth curves and only constraints in one dimension are considered; applying this multi-stage casting process for fabricating objects with more intricate shapes is more difficult. StackMold works with complex shapes and multiple materials, computationally optimizing the mold design and offering step-by-step instructions to the user. We outline the algorithms involved in this optimization in the implementation section.

## 3.3 System Overview

### 3.3.1 Processing 3D Models

StackMold operates on pre-existing 3D models, readily available online from websites such as Thingiverse<sup>1</sup>. A user imports a model into the StackMold software and uses 3D brushes to annotate which regions of the model should be cast in the desired materials. The user can also place external components such as electronics on the model. After the user specifies the thickness of the material they will use to create the laser-cut mold slices, StackMold presents a preview of the to-be-cast object.

### 3.3.2 Casting High-Fidelity Surface Regions

By default, the cast object, and therefore preview, will be blocky and coarse: a result of building the mold from slices of laser-cut material. StackMold offers three techniques to enable the user to preserve surface details.

First, users can adjust the cast orientation of the object. Our sliced mold buildup offers highly detailed contours within a slice, while the object is coarse across slices. Manipulating the casting orientation therefore controls the directions in which contours are coarse and detailed.

Second, users can choose a different mold material thickness. While a thinner material preserves more contour details across layers, it involves laser cutting more slices, increasing both fabrication time and the consumption of material.

Finally, StackMold supports 3D printing parts of the mold to allow for regions in which contour details are preserved in all directions. During the fabrication stage, users annotate desired high-fidelity surface regions using a 3D brush. For those regions, StackMold generates 3D-printed parts that fit in the laser cut mold slices. In contrast to existing computational molding approaches that 3D print the entire mold [1, 15], our approach is significantly faster: we print only for desired high-fidelity regions, and because the laser-cut mold offers support, the printed parts are thin, fast-printing shells.

### 3.3.3 Hollow and Multi-Material Volumes Inside Objects

In addition to specifying material properties of a surface mesh, StackMold also allows for specifying material properties of sub-volumes inside objects. This is supported by importing additional mesh sub-volumes and positioning them inside the object. In the same way as users can annotate the main model, StackMold supports detailed sub-volume material configuration using the 3D surface brushes.

Similar to 3D printing technologies, overhanging structures require a support structure before casting. Overhanging structures outside the object are supported by mold layers as shown in Figure 3.5a. In contrast, overhanging structures above hollow volumes inside the object, such as the top of the vase in Figure 3.5b, cannot be supported by the layered mold

---

<sup>1</sup><https://thingiverse.com>



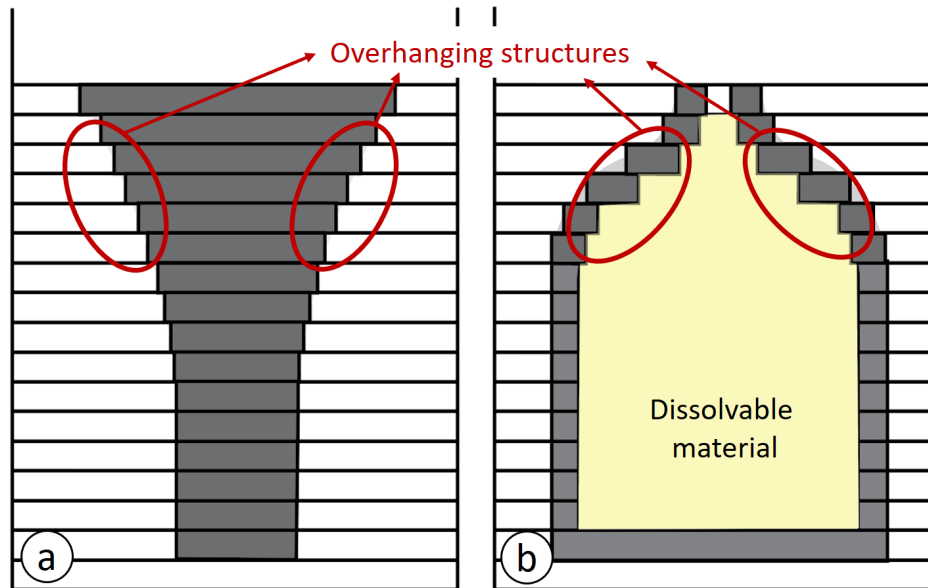


FIGURE 3.5: (a) Overhanging structure supported by mold layers. (b) Internal overhanging structure supported by dissolvable material.

structure. Furthermore, temporary props cannot support the overhanging structure in this example, as the object’s contour prevents the removal of these props afterwards. In these situations, StackMold requires temporarily filling the hollow region with dissolvable material (Figure 3.5b). In our examples, we use beeswax as it is easy to cast and convenient to dissolve using heat. Alternatively, water dissolvable materials could be used, but these materials often require cavities to be accessible by a water jet to properly remove the material.

### 3.3.4 Laser Cutting and Labeling Parts

All slices of the mold layers are exported to an SVG file for laser cutting (Figure 3.6a). This SVG also includes the alignment box, brackets, and engraved labels for all parts for easy reference in the step-by-step manual. Although the laser cut parts in the actual material volumes are taken out before casting, some of them are used as temporary props and are therefore labeled. Laser cut parts that are too small to fit an identifier are added to a “small part catalog”. The contours of all parts in this catalog are printed on a sheet of paper to facilitate their identification by matching the laser cut shape on the sheet (Figure 3.6b).

### 3.3.5 Step-By-Step Instruction Manual

StackMold compiles a web page with a step-by-step instruction manual. Figure 3.3 shows a set of instructions for casting a multi-material audio headset. The manual refers to the mold layers and temporary props using their unique identifier, which is laser engraved on the part or for very small parts can be looked-up by matching the laser cut part on a printed stencil of small parts (Figure 3.6b). As StackMolds oftentimes expose multiple compartments at the same time, the manual visually highlights the appropriate compartment during casting operations (blue circle in Figure 3.3). Although StackMold computes the amount of casting



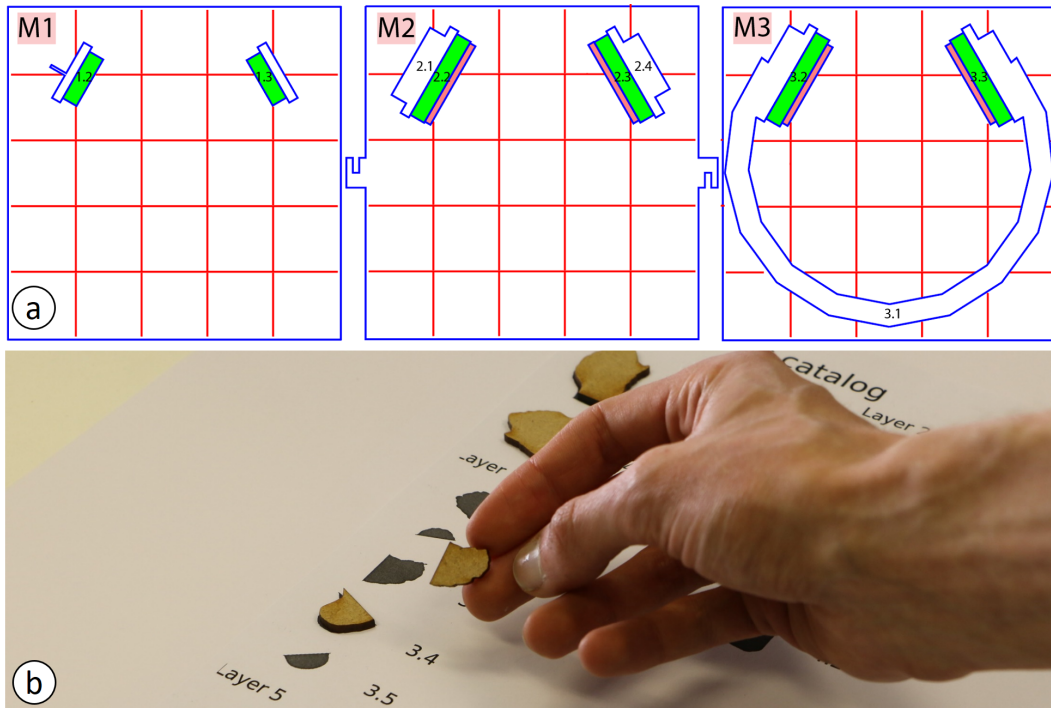


FIGURE 3.6: (a) Laser cut features have identifiers engraved. (b) Identifiers of very small parts are looked up in a “small part catalog”

material required to fill every compartment, relying solely on this measure is not very accurate, as some residual material always remains in the mixing cup. Our multi-stage mold buildup therefore ensures that molds can always be fully filled in every casting step. We refer the reader to the supplementary material, attached to this submission, for a full instruction manual for making a multi-material turtle toy character using StackMold.



## 4 Software Implementation

StackMold’s software environment is implemented as a cross-platform Electron app<sup>1</sup> in JavaScript, primarily using three.js<sup>2</sup>, a library for programming with WebGL. In this section, we discuss the algorithms and computational techniques for designing a multi-stage mold.

### 4.1 Multi-Material Slicing

Triggering StackMold’s fabrication stage (step 2 in walkthrough) renders a preview of the final multi-material object in real-time. The object is first sliced according to the user-specified mold-material thickness. This slicing involves calculating the intersections between a plane at the center of every slice and the faces in that region. These intersection points define the contour of each slice. StackMold suggests reducing the slicing thickness when the slicing procedure introduces a non-manifold mesh or splits the object in disconnected volumes (Figure 4.1).

Faces with material annotations different from the base-material represent 3D volume extrusions inside the object. Similar to Brochu et al. [5], this volume is defined by either connecting the end points of the 3D-annotated surface or by the intrusion-depth of the annotation brush. Both settings are available for all material brushes in the StackMold interface, but many alternative volume selection techniques could be supported in the future, such as 3D lasso tools [27]. Slices consisting of multiple materials are split in separate material volumes. Depending on the brush settings, this is done by either connecting the end points or by eroding the sampled contour of the annotated faces using the intrusion depth of the brush.

### 4.2 Venting Pipes

As in any casting process, air bubbles can get trapped in cavities during the casting process. Similar to FlexMold [15], we resolve this issue by analyzing the model for local maxima. We laser cut a small circle in all mold layers above each maximum, forming vertical pipes that allow air to escape during the casting process.

---

<sup>1</sup><https://electronjs.org>

<sup>2</sup><https://threejs.org>

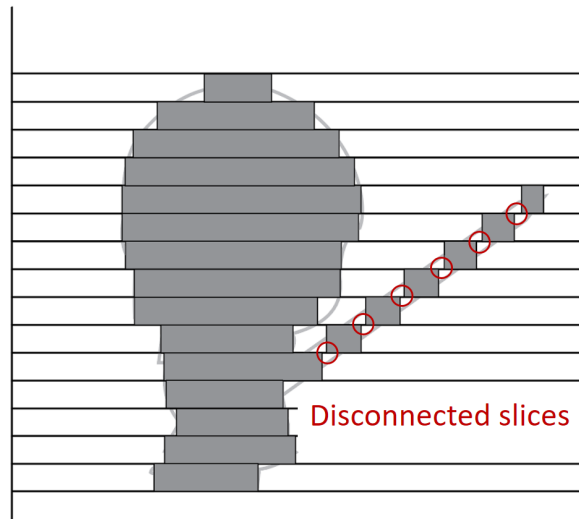


FIGURE 4.1: StackMold suggests reducing the slicing thickness when parts of the volume are disconnected.

### 4.3 Generating 3D printed parts

For faces annotated as high-fidelity surface regions, StackMold generates STL files for 3D printing those parts of the mold. Both the SVG and STL files require adjustments. The laser cut mold layers need to provide space for the 3D printed part and the 3D printed part has to fit in the stair-case mold structure.

Our approach is shown in Figure 4.2. (a) We first create a surface shell of the selected faces by taking a copy of those faces. (b) The surface shell is extruded 5mm in the direction of the normal of the faces to realize a manifold mesh. Next, the slicing technique, discussed in section “*Multi-Material Slicing*”, is applied to re-sample the contour of the mold layers to make space for the 3D surface mold. (c) The original surface shell is now further extruded (10mm) to ensure the 3D surface mold intersects, and thus fits, all slices. (d) Finally, the sliced mold layers are subtracted from the 3D surface mold to realize the final 3D printable mold. This 3D mold perfectly fits the surrounding laser cut mold and attaches firmly with a drop of superglue. To easily remove the 3D printed molds after casting, we print 3D mold parts with flexible filament (Ultimaker TPU95A).

### 4.4 Computing viable cast sequences

Our computational approach optimizes the multi-stage mold buildup and minimizes the number of casting and curing steps. In the most optimal cast sequence, the number of casting steps equals the number of different materials in the object. However, even for simple designs this is not always feasible, as shown in Figure 2.13. First, we calculate viable cast sequences starting from a representation in which all material volumes per layer are considered separate compartments (Figure 4.3a). From this representation, we calculate which volumes can be cast at the same time. This process starts with initiating a *volume dependency table* which links every material volume to the volumes below that are in direct contact (Figure 4.3b). For

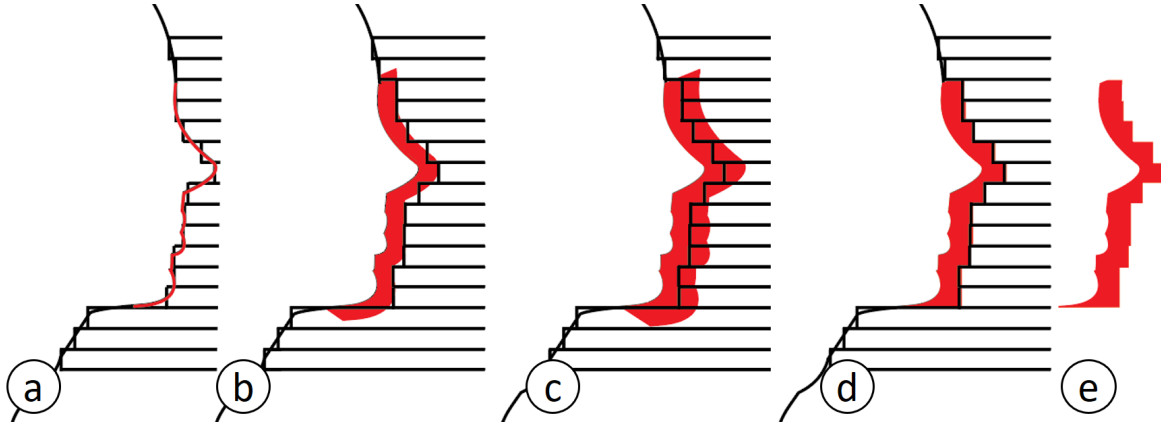


FIGURE 4.2: StackMold generates 3D-printed mold pieces by (a) copying user-specified faces from the original 3D model; (b) extruding to form an initial mold and re-sampling the slices to the region; (c) the 3D mold is further extruded to intersect all mold slices; (d-e) subtracting the slices from the 3D mold to form a “stairstep” pattern.

example, A2 and B2 both have A1 as a dependency, meaning that both of them cannot be cast before A1. In contrast, material volume A3 could be cast before B2, as they do not have a dependency. In contrast to the simplified 1D example in Figure 4.3a-b, our implementation considers overlaps between the polygonal material volumes in two dimensions.

Using the *volume dependency table*, viable cast sequences are calculated and modeled in a *viable cast sequences tree* (Figure 4.3c). Every path from the root to a leaf is considered a viable cast sequence for fabricating the object. The sequence A1-A2-A3-A4-B2-B3 seems the most optimal solution, as it only requires one casting step for every material. However, this is an invalid solution, as it requires positioning of temporary props in B compartments, of which some cannot be removed after casting the A material volumes. More specifically, the temporary prop in B2 cannot be removed, as it is blocked from below by A1 and on top by mold layer M3, which in turn is blocked by material volume A4. To validate and prune the large solution space of cast sequences, the next section covers the algorithm for modeling the positioning and removal of temporary props and mold layers.

## 4.5 Validity analysis and pruning of cast sequences

A valid mold buildup ensures that isolated compartments are formed during the casting process to support individually casting each desired material. This requires compartments to be accessible for casting and timely placement and removal of mold layers and temporary props. Our system computes the mold buildup and its validity while computing the *viable cast sequences tree* in Figure 4.3c. Algorithm 1 calculates and validates the multi-stage mold buildup for a given cast sequence. This function is called every time an element is added to the *viable cast sequences tree*. Running this algorithm for partial cast sequences allows for pruning large chunks of the tree early, as the algorithm identifies an invalid mold buildup (*InvalidSequenceException*). Algorithm 1 takes as input the array elements (material volumes) starting from the root of the *viable cast sequences tree* to the element last added. It uses

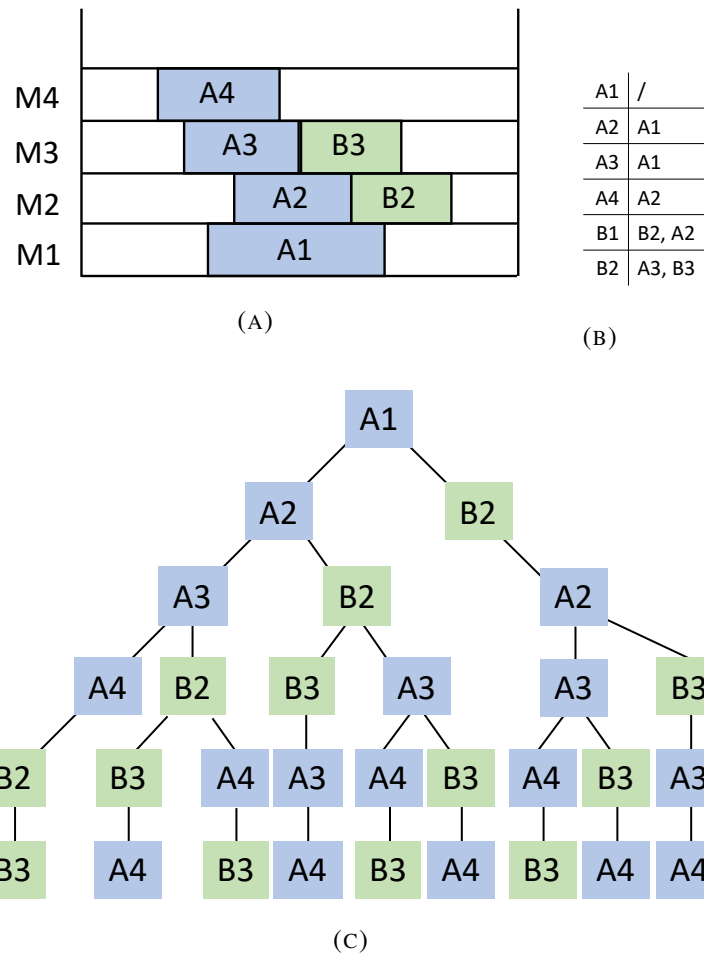


FIGURE 4.3: (a) A simplified representation of four layers to be cast in two materials, A and B. (b) A *volume dependency table* for the example. (c) The *viable cast sequence tree* for the example, showing different possible sequences for casting.

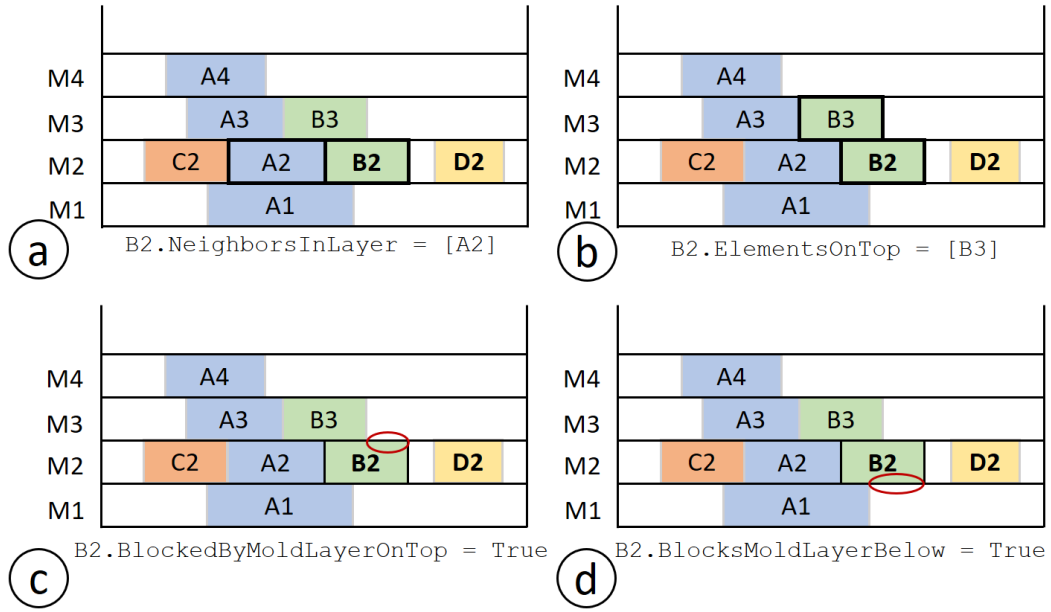


FIGURE 4.4: Properties stored by StackMold for each casting volume, using B2 as the example material volume: (a) material volumes in the same layer which touch B2; (b) volumes touching from above; and if the material volume touches the mold itself (c) above or (d) below.

Algorithms 2 and 3 to compute and validate the required mold buildup. Algorithm 1 computes and returns all casting and assembly steps, including the positioning and removal of temporary props and mold layers. StackMold considers the cast sequence with the lowest number of casting/curing steps as the most optimal solution.

The algorithms are called continuously while optimizing the cast sequence. Therefore, the geometric and spatial relations between material volumes and mold layers are calculated beforehand and stored as properties in all material volumes of the input array. As shown in Figure 4.4, every material volume stores four geometric properties: (a) the neighboring material volumes in the same layer that are in direct contact; (b) the material volumes in the layer on top that are in direct contact; (c) whether the material volume is in direct contact with, and thus blocked by, the mold layer on top; (d) whether the material volume is in direct contact with, and thus is blocked by, the layer below. In contrast to the simplified 1D representation in Figure 4.4, our implementation considers these geometric properties in 2D, as the material volumes are defined by polygons.

In addition to pruning invalid partial solutions of the *viable cast sequences tree*, our implementation also uses heuristics for additional pruning of the tree. This includes processing material volumes with the same material properties first and prioritizing material volumes that are connected across mold layers. This heuristic allows for finding a good solution (limited number of casting/curing steps) early and further prune the solution space.

Despite these pruning steps, the solution space can still be very large for advanced examples consisting of many layers and different material volumes per layer. However, for the advanced examples in this paper (see “Example Designs” section), our algorithm always finds a good solution within several seconds. While the algorithm continues to optimize

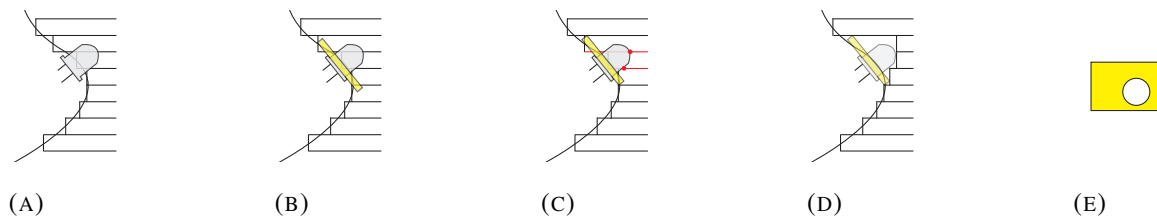


FIGURE 4.5: Embedding an electronic component into the cast object, using an LED as an example. (a) The user positioned the component on the to-be-molded object. StackMold then (b) places its parametric fixture and (c) sizes it to fit within the intersecting mold layers, (d) adjusts the intersecting layers to prevent interference with the component and fixture, and (e) generates the final fixture.

the solution further, the user can accept at any time the best solution available. When the algorithm cannot find an optimal solution or any solution at all in a reasonable amount of time, the user is offered the option to compute a sub-optimal solution. A sub-optimal solution is computed by applying the algorithm, outlined in this section, to a subset of mold layers at a time. This approach is significantly faster, as it optimizes the buildup of groups of layers and thus approximates the optimal solution in which the buildup of all layers is optimized. Oftentimes, the sub-optimal solution involves more casting/curing steps compared to the optimal solution.

## 4.6 Component Integration

To embed existing components, such as electronics, in the surface region of an object, users position the electronic component by specifying its location on object's surface. The component is aligned with the average orientation of the normals of the faces in that region. During the generation of the multi-stage mold, fixtures are added to hold the component in place during the casting. These fixtures are designed to precisely fit the component and have a parametric shape to fit the slices of the mold layers. The current version of StackMold supports embedding a predefined set of electronic components, including an LED, a switch, a button, and speakers.

Figure 4.5 illustrates the approach to fit fixtures in the mold layers, using the example of an LED. The parametric fixture for the LED is positioned and oriented according to the component's position (a–b). The size of the fixture is adjusted to the slicing interval of the mold layers (b–c). Intersections between mold layers, the component, and the fixture are calculated and the sampled mold layers are adjusted to avoid overlaps (d). The final fixture is exported for laser cutting (e) and attaches to the MDF mold layers using a drop of superglue.

While fabricating fixtures with a laser cutter is fast, surface details are lost because of the flat fixture design. Alternatively, the component is positioned in a high-fidelity mold region produced with 3D printed parts. In this case, the 3D mesh of the existing component is subtracted from the generated 3D mold part (see section “Generating 3D printed parts”) to realize holes for fixing the component.



---

\*TODO add example where we go through the build up of the tree and when what decisions are made with images to give the reader a good understanding of how all the algorithms work together in practice.\*

**Algorithm 1:** computeAllInstructions

---

```

1 lastLockedLayerID  $\leftarrow$  0
2 lastMoldLayerID  $\leftarrow$  0
3 castElementsAtOnce  $\leftarrow$  new Array()
4
   input : Array of elements from the root to a leaf (seq). Every element has the following
           properties: LayerID, MaterialID, IsTempProp, Iscast, NeighborsInLayer,
           BlocksMoldLayerBelow, ElementsRestingOnTop, BlockedByMoldLayerOnTop
   output : Array of instructions, including mold assembling and casting (allInstructions)
5 Function computeAllInstructions(seq)
6   allInstructions  $\leftarrow$  new Array()
7   foreach  $el \in seq$  do
8     lastItem  $\leftarrow$  castElementsAtOnce.last()
9     if !connectedWithSameMaterial( $el$ , lastItem) then
10      | castCollectionOfElements()
11    end
12
13    if  $el.IsTempProp$  then
14      | tryToRemoveProp( $el$ )
15    end
16
17     $u \leftarrow$  moldLayerAssembleInstru(lastMoldLayerID,  $el.LayerID$ )
18    allInstructions.append( $u$ )
19    lastMoldLayerID  $\leftarrow$   $el.LayerID$ 
20
21    foreach  $n \in el.NeighborsInLayer$  do
22      | if ! $n.Iscast$  and ! $n.isTempProp$  then
23        |  $n.isTempProp \leftarrow$  True
24        |  $u \leftarrow$  makeInstrucToAddTempProp( $n$ )
25        | allInstructions.append( $u$ )
26      | end
27    end
28
29    castElementsAtOnce.push( $el$ )
30  end
31
32  castCollectionOfElements()
33  return allInstructions

```

---

**Algorithm 2:** castCollectionOfElements

---

```

1 Function castCollectionOfElements()
2   prevEl  $\leftarrow$  castElementsAtOnce.last()
3   if prevEl.LayerID < lastLockedLayerID then
4     | Throw InvalidSequenceException()
5   end
6
7   if prevEl.LayerID < lastMoldLayerID then
8     | u  $\leftarrow$  instucRemoveLayers(prevEl.LayerID, lastMoldLayerID)
9     | allInstructions.append(u)
10    | lastMoldLayerID  $\leftarrow$  prevEl.LayerID
11  end
12
13  foreach castEl  $\in$  castElementsAtOnce do
14    | castEl.IsCast  $\leftarrow$  True
15    | if castEl.BlocksMoldLayerBelow
16    |   and castEl.LayerID - 1 > lastLockedLayerID then
17    |   | lastLockedLayerID  $\leftarrow$  castEl.LayerID - 1
18    |   end
19  end
20
21  foreach fixture  $\in$  fixtures do
22    | if fixture in area castElementsAtOnce then
23    |   | u  $\leftarrow$  makeInstructionToAddFixture(fixture)
24    |   | allInstructions.append(u)
25    |   end
26  end
27
28  u  $\leftarrow$  makeCastInstructions(castElementsAtOnce)
29  allInstructions.append(u)
30  castElementsAtOnce  $\leftarrow$  new Array()

```

---

**Algorithm 3:** tryToRemoveProp

---

```

1 Function tryToRemoveProp(element)
2   if element.BlockedByMoldLayerOnTop then
3     nextID  $\leftarrow$  element.LayerID + 1
4     if nextID  $\leq$  lastLockedLayerID then
5       Throw InvalidSequenceException()
6     end
7
8     u  $\leftarrow$  instucRemoveLayers(nextID, lastMoldLayerID)
9     allInstructions.append(u)
10    lastMoldLayerID  $\leftarrow$  element.LayerID
11  end
12
13  foreach elTop  $\in$  element.ElementsRestingOnTop do
14    if elementOnTop.IsTempProp then
15      tryToRemoveProp(elementOnTop)
16    else
17      |
18    end
19  end
20
21  foreach fixture in fixtures do
22    if fixture.isPlaced and element.isBlockedBy(fixture) then
23      u  $\leftarrow$  makeInstructionToRemoveFixture(fixture)
24      allInstructions.append(u)
25    end
26  end
27
28  element.IsTempProp  $\leftarrow$  False
29  u  $\leftarrow$  removePropInstruction(element)
30  allInstructions.append(u)

```

---

## 5 Example Designs

Using StackMold’s software environment, 4 example objects shown in Figure 5.1 were fabricated: (a) a soft turtle toy character with different colors of silicone, (b) a multi-colored silicone figurine of which the head is cast in high-fidelity using 3D printed mold parts, (c) a vase made of plaster of which the overhanging structure requires embedding dissolvable material (beeswax) during the fabrication process, (d) a functional audio headset consisting of a rigid resin headband and soft silicone earpads with embedded speakers.

Figure 5.1 together with chapter 5 also reports the number of curing steps for every example object and the total fabrication time. Fabricating these five objects low resolution with a single material using an Ultimaker 3 FDM printer would take respectively take 5:20, 5:09, 2:20, and 6:05 hours. Note that although we compare the fabrication times with StackMold to an FDM 3D printer, most of our examples cannot be fabricated with conventional FDM printers, including the Ultimaker 3, as they do not allow for more than two materials and embedded electronics. Also note that fabricating the plaster model with StackMold takes longer compared to 3D printing as plaster has to dry for several hours. Molding large objects in plaster, however, would be faster compared to 3D printing as the curing time does not increase with the material volume.

Object	FDM	StackMold		
	print time	Total time	Curing steps	Curing time
(a) Turtle	5:20	2:00	3	0:30
(b) Figurine	5:09	1:30	3	0:30
(c) Plaster vase	2:20	4:30	3	3:30
(d) Headphones	6:05	2:00	2	0:20

TABLE 5.1: The time taken to print the object with an FDM printer compared to our method. The purpose of this table is here to provide a general idea of how long the casting takes because this comparison is like comparing apples to oranges. Most FDM printers can not print more than two materials in one print and does not have support for electronics, while our method requires more manual labor and does not have a high fidelity.

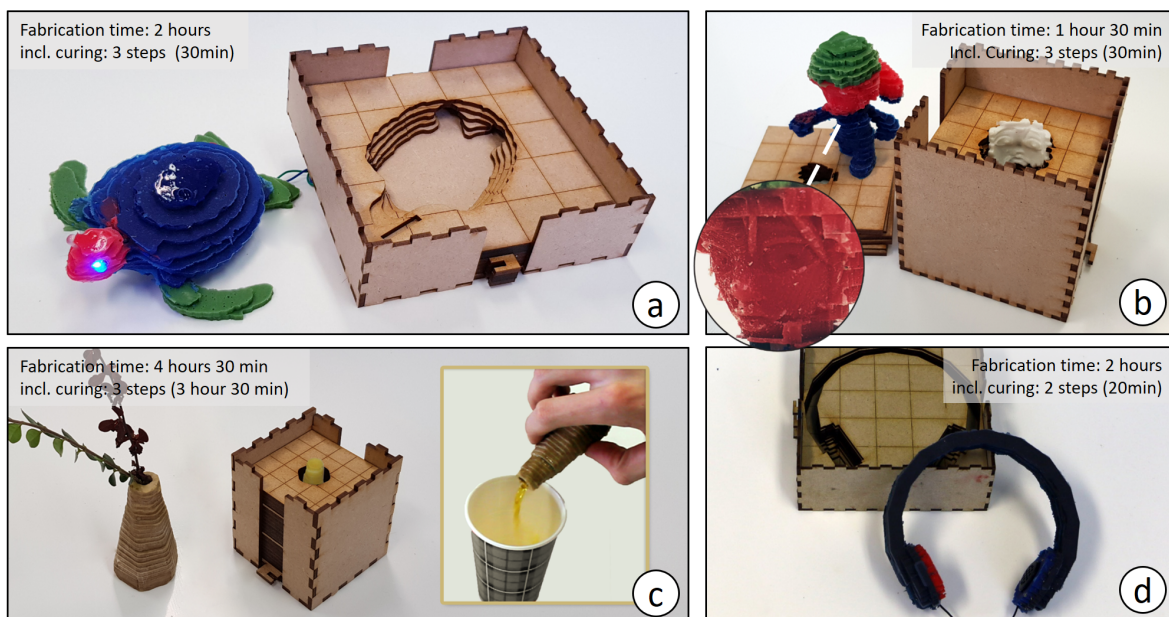


FIGURE 5.1: Example objects fabricated with StackMold: (a) a silicone turtle character with different colors, (b) a multi-colored silicone figurine with regions in high-fidelity, (c) a vase made of plaster using dissolvable material, (d) a functional audio headset consisting of resin and silicone.

## 6 Limitations and Future Work

Although StackMold offers many novel opportunities for multi-material fabrication and rapid prototyping of functional objects, our technique also has limitations which reveal many interesting challenges for future research.

First, StackMold computationally optimizes the number of casting and curing steps, but there are still some objects that could require many casting steps. One class is objects with intricate multi-material compositions, involving either many materials, such as material gradients (Figure 6.1a), or exposing separate material volumes with the same material properties, such as checkerboard patterns (Figure 6.1b). Another class of objects requires many casting steps because of their surface geometry. For example, casting the Christmas tree shown in Figure 6.1c in two materials requires positioning temporary props in one half of the tree. The narrow regions prevent the removal of those props and thus requires every branch of the tree to be cast separately. Changing the cast orientation could significantly reduce the number of casting steps for some objects. Alternatively, future versions of StackMold could support mold layers split in multiple jig saw puzzle parts. Using this approach, mold layers and temporary props could potentially be removed sideways without removing all mold layers on top or below. Additionally, this approach could allow for reusable multi-stage molds, as those molds are easier to disassemble.

Second, StackMold could be extended with various computational features to further increase the surface fidelity of cast objects. For example, the slicing thickness could dynamically adjust to fit the level of surface details in regions of the object. Thin slices preserve surface details for intricate curves, while coarse slices are sufficient for simple contours like in Figure 6.3. Additionally, regions of the mold could be sliced in different orientations to preserve surface details in different directions across the object. For example, slicing the body of a figurine horizontally and the head vertically within a single multi-stage mold. Figure 6.2 shows such a sliced example. It would result in a very detailed head when viewed from the front and a very detailed body when viewed from above.

Third, the current version of StackMold supports embedding existing components on the surface of objects, but not inside the object. Embedding components inside a multi-stage mold requires more considerations, as these components could prevent positioning and removing temporary props and mold layers, as well as the access to casting compartments (Figure 6.4). With mechanical components like axles or gears, there is the extra requirement of freeing the surrounding area from casted material. An axle should be free to rotate but when material surrounds the object, it is not able to do that.

Last, some casting materials are hard to combine or adhere. Aggressive resins could disintegrate silicone while curing. Future versions of StackMold can consider these characteristics when optimizing the order of cast operations. This could have a negative impact on

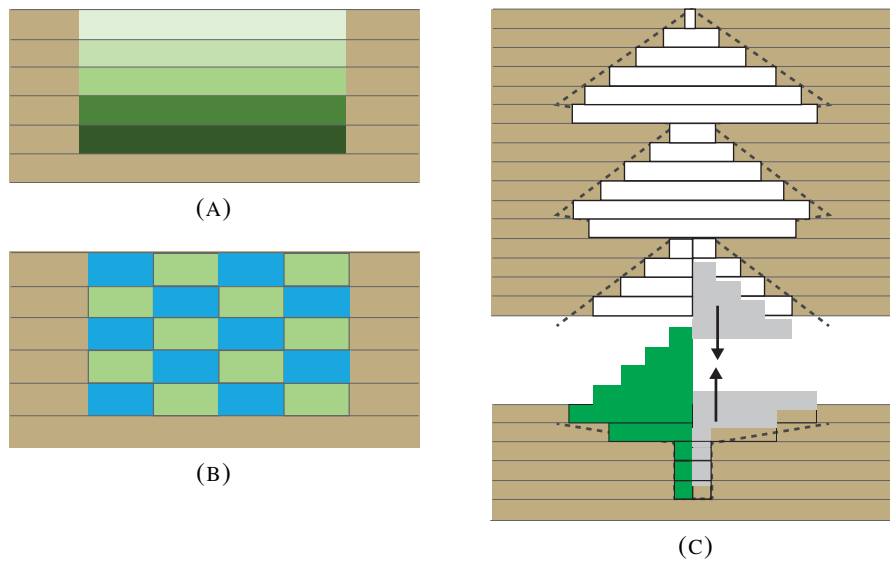


FIGURE 6.1: Some limitations of StackMold: objects with many material changes such as gradients (a) or patterns (b) will require a large number of casting steps. Objects like the tree in (c) require more steps in order to remove temporary props.

the number of casting steps. Besides this, adhering silicons to other materials is sometimes challenging. Clamp designs could be positioned at junctions to mechanically interconnect materials (Figure 6.5). Holes for these clamps are created in one material and the liquid cast of the other material will then flow into the clamp shape, ensuring a very tight fit. Despite changing the form of some placeholders and inside of material layers, it does not change anything to the current process which is a plus. However, clamps could add extra dependencies and thus add casting steps to a model. Therefore the placement, orientation, size and shape of these clamps must be carefully calculated to minimize their effect on the build time. It is important to note that clamps are not only important in a 2D plane but also in a 3D space. The feet of the turtle in Figure 6.5 are very strong when they are pulled sideways or upwards but when the leg is pulled downwards it can be removed, while the clamp on the turtle head is fully surrounded and stays firmly attached to the model.

## 6.1 summary

In summary what challenges are solved.

1. *Support for a broader range of materials.* - Stackmold theoretically supports all castable materials. This on itself is very promising and allows for some material combinations for prototypes that previously required more manual labor from the maker. However *theoretically* needs to be emphasized here. Stackmold does not solve any issues regarding materials that do not naturally adhere to each other nor problems surrounding combining materials that possibly damage each other while casting.



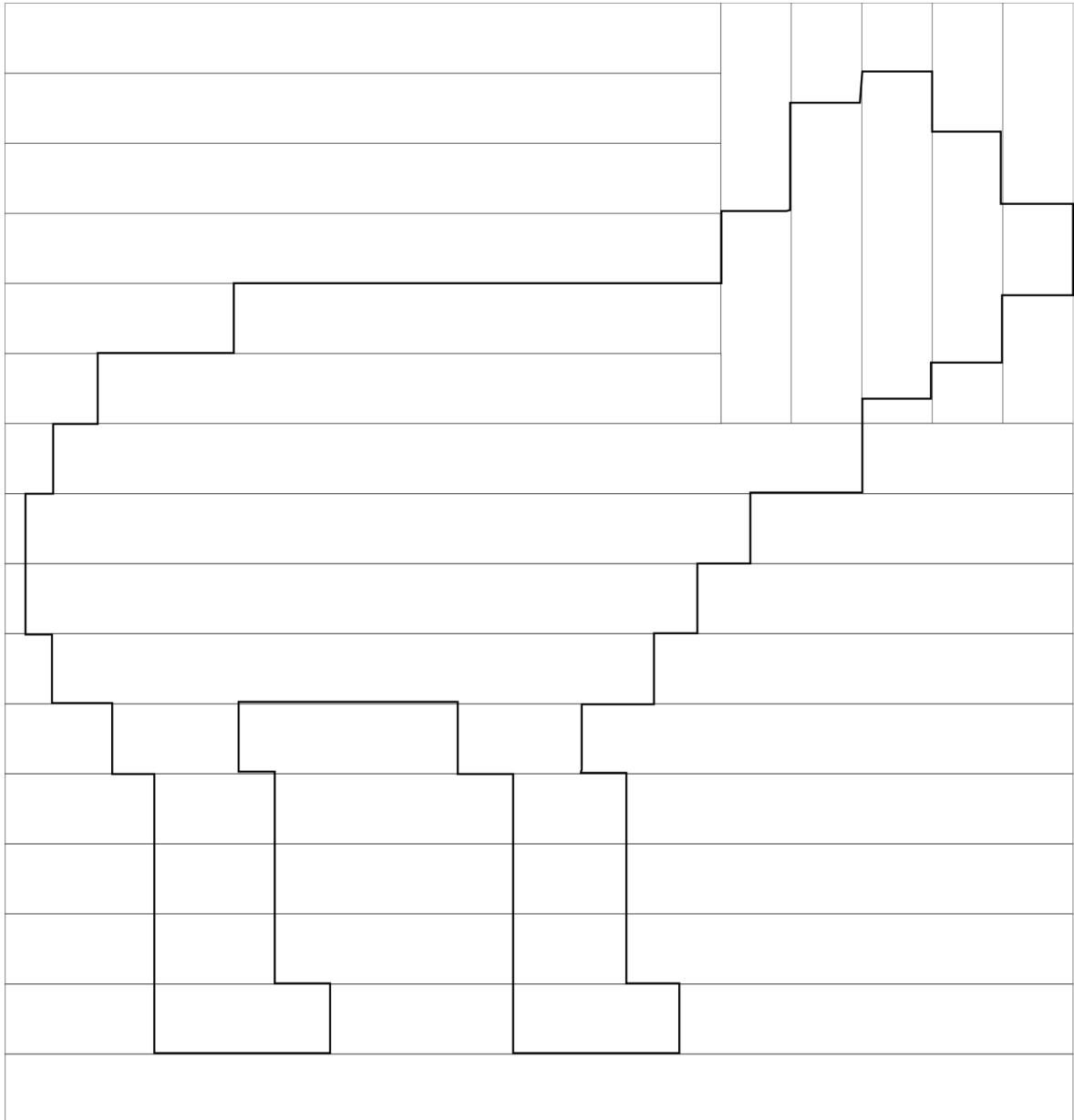


FIGURE 6.2: Different slice orientations could be future work. The head of this figurine is sliced in a vertical direction while the rest of the body is sliced horizontally which result in a very detailed head when viewed from the front and a detailed body when viewed from the top.

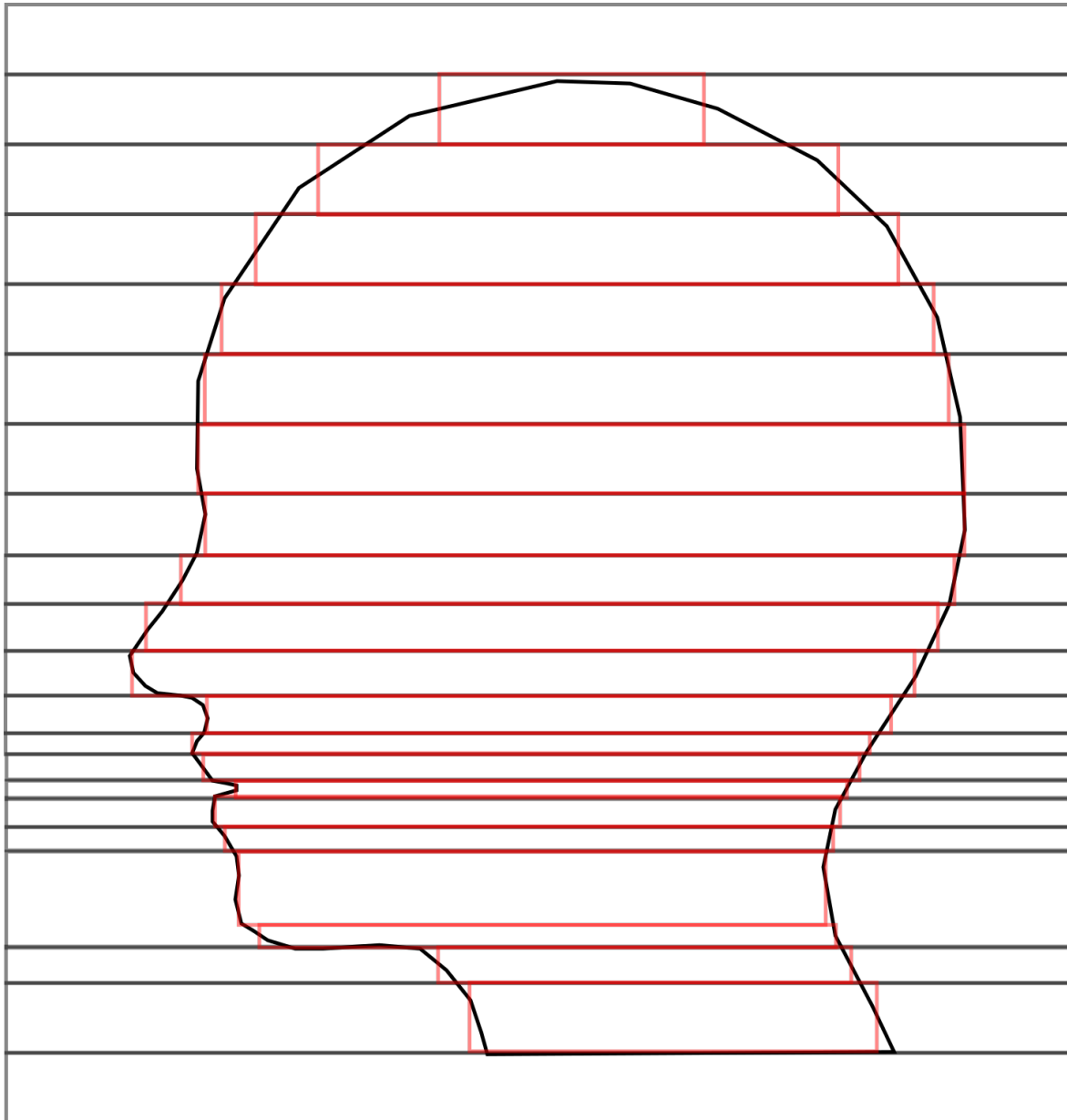


FIGURE 6.3: A head sliced with dynamic layer thicknesses. Makes layers thinner in tight corners to preserve surface detail. Idea for future work.

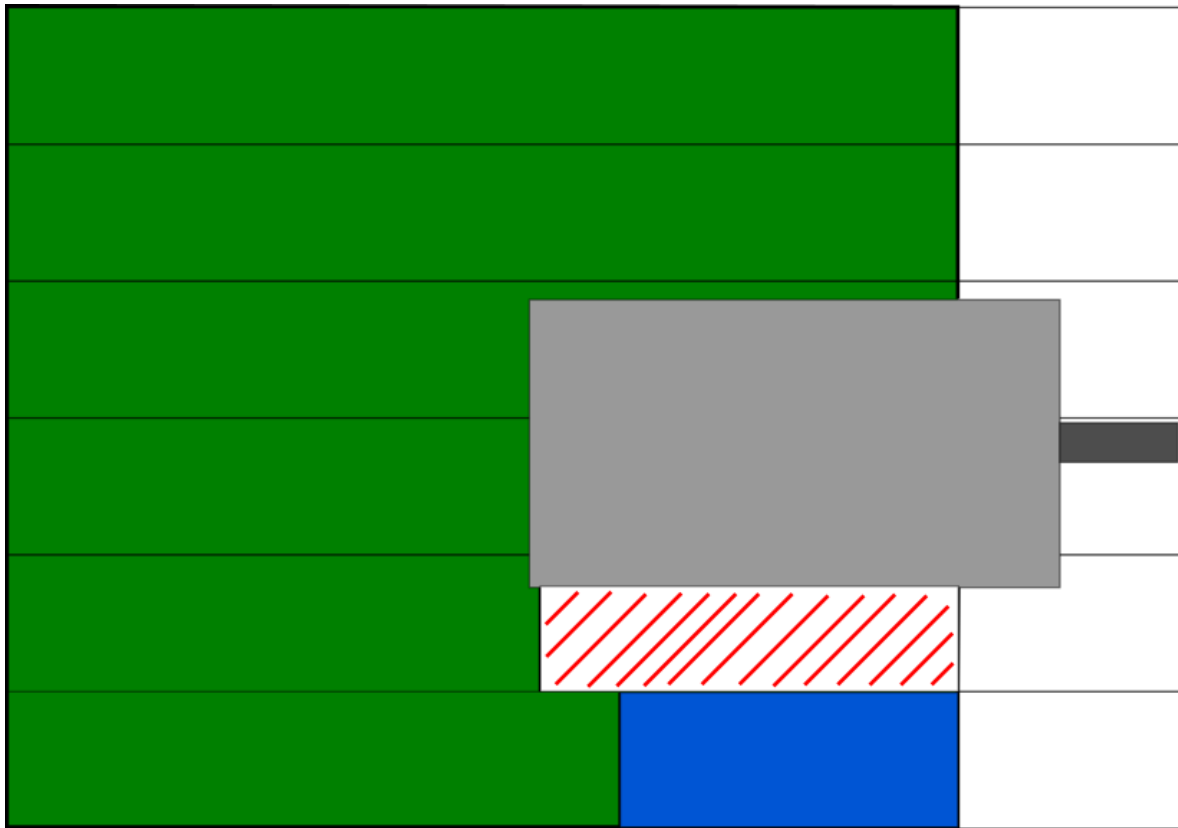


FIGURE 6.4: With the motor in this position, there is no possibility for the red striped region to be casted.

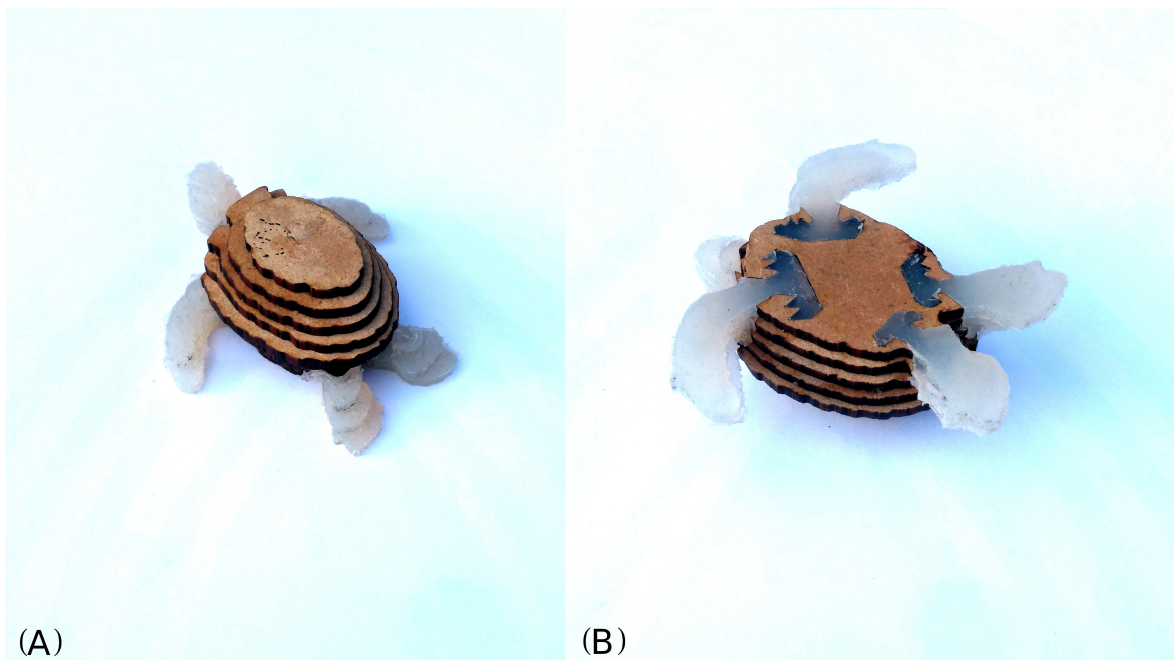


FIGURE 6.5: A turtle model with an mdf shell and silicon legs and head (a), same model turned upside down (b). The clamps ensure that the silicone stays attached to the mdf. The head is attached in the same way however this is not visible from the outside.

2. *Being able to embed electronics and mechanical parts into the object without the need for the creator to adjust the model by him or herself.* - The functionality of adding electronic components to the models surface is a step into the right direction and could certainly make the prototyping of devices that rely on small surface mounted electronics substantially easier. However, in regards to embedding electronics and mechanical parts inside the mold, our approach does not add any benefit for the maker.
3. *A higher fabrication speed. The faster a prototype can be created, the more iterations you can do over them.* - Although a comparison between 3D printing and this technique is comparing apples to oranges on quit average settings for an FDM printer, Stackmold outperforms the 3D printer rather easily in time. This is definitely a step in the right direction.
4. *Prototyping should be more easier for a novice user. Less requirements to start, less experience needed and a more affordable process.* - The maker loads in a model, annotates the material and then follows an instruction manual to fabricate the device. In this routine the maker did not have to design a separate model for each of the different materials, he did not have to think about the best order to the put the different compartments together and he did not need to design mounting holes for the surface electronics. Stackmold only requires access to a lasercutter in a makerspace for example. Casting materials are also mostly less costly then 3D printer filament. Thus StackMold solves many issues and makes the prototype process overall easier, more accessible and affordable. However the user still needs to know how certain materials will affect each other, like resins and silicone for example. and also how to hold them together.
5. *Planning on when to use what material and when to add it. Molten metals might melt polymers when they are added in the wrong order.* - As already explained, despite some exploration into this problem(e.g. clamps) this thesis does not provide any help in dealing with this problem.

## 7 Conclusion

In this Master Thesis, we presented StackMold, a novel multi-stage molding technique to realize multi-material and multi-colored objects. To support users making and using StackMolds, we contributed algorithms to computationally design multi-stage molds and an end-to-end software environment to guide users through the building and molding process. StackMold contributes in parallel to multiple active research challenges in personal fabrication, including multi-material fabrication, techniques to speed-up fabrication, and fabricating functional electronic objects. Therefore our fabrication process can be used during many stages of a prototyping process; for building low-fidelity coarse representations as well as for realizing high-fidelity functional objects. Our work also opens many opportunities for future work. One aspect that is especially appealing is the ease with which fully tested electronic circuits can be seamlessly embedded. This is in contrast to existing throw-away prototyping processes which require users to start on breadboards and later move to PCBs or novel conductive inkjet printing processes to seamlessly embed circuits in objects.



# Bibliography

- [1] Thomas Alderighi, Luigi Malomo, Daniela Giorgi, Nico Pietroni, Bernd Bickel, and Paolo Cignoni. 2018. Metamolds: Computational Design of Silicone Molds. *ACM Trans. Graph.* 37, 4, Article 136 (July 2018), 13 pages. DOI:<http://dx.doi.org/10.1145/3197517.3201381>
- [2] Callum Bailey, Efrain Aguilera, David Espalin, Jose Motta, Alfonso Fernandez, Mireya A Perez, Christopher Dibiasio, Dariusz Pryputniewicz, Eric Macdonald, and Ryan B Wicker. 2018. Augmenting computer-aided design software with multi-functional capabilities to automate multi-process additive manufacturing. *IEEE Access* 6 (2018), 1985–1994.
- [3] Dustin Beyer, Serafima Gurevich, Stefanie Mueller, Hsiang-Ting Chen, and Patrick Baudisch. 2015. Platener: Low-Fidelity Fabrication of 3D Objects by Substituting 3D Print with Laser-Cut Plates. In *CHI '15: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*.
- [4] Bernd Bickel, Moritz Bächer, Miguel A. Otaduy, Hyunho Richard Lee, Hanspeter Pfister, Markus Gross, and Wojciech Matusik. 2010. Design and Fabrication of Materials with Desired Deformation Behavior. *ACM Trans. Graph.* 29, 4, Article 63 (July 2010), 10 pages. DOI:<http://dx.doi.org/10.1145/1778765.1778800>
- [5] T Brochu and Ryan Schmidt. 2017. Geometric Modeling of Multi-Material Printed Objects. In *Eurographics Short Papers 2017*.
- [6] X Chen, H Li, C W Fu, H Zhang, D Cohen-Or SIGGRAPH Asia 2018, and 2018. 2018b. 3D fabrication with universal building blocks and pyramidal shells. *dl.acm.org* (2018).
- [7] Xiang 'Anthony' Chen, Stelian Coros, and Scott E Hudson. 2018a. Medley: A Library of Embeddables to Explore Rich Material Properties for 3D Printed Objects. In *CHI '18: Proceedings of the 36th Annual ACM Conference on Human Factors in Computing Systems*. ACM, New York, New York, USA, 162–172.
- [8] Jae-Won Choi, Ho-Chan Kim, and Ryan Wicker. 2011. Multi-material stereolithography. *Journal of Materials Processing Technology* 211, 3 (2011), 318–328.
- [9] Paolo Cignoni, Nico Pietroni, Luigi Malomo, and Roberto Scopigno. 2014. Field-aligned mesh joinery. *ACM Transactions on Graphics (TOG)* 33, 1 (Jan. 2014), 11–12.
- [10] Leen Danny, Veuskens Tom, Kris Luyten, and Raf Ramakers. 2019. JigFab: Computational Fabrication of Constraints to Facilitate Woodworking with Power Tools. In

- Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '19)*. ACM.
- [11] Claudia Daudén Roquet, Jeeun Kim, and Tom Yeh. 2016. 3D Folded PrintGami: Transforming Passive 3D Printed Objects to Interactive by Inserted Paper Origami Circuits. In *Proceedings of the 2016 ACM Conference on Designing Interactive Systems (DIS '16)*. ACM, New York, NY, USA, 187–191. DOI:<http://dx.doi.org/10.1145/2901790.2901891>
- [12] Laura Devendorf and Kimiko Ryokai. 2015. Being the Machine: Reconfiguring Agency and Control in Hybrid Fabrication. In *CHI '15: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM Press, New York, New York, USA, 2477–2486.
- [13] Philipp Herholz, Wojciech Matusik, and Marc Alexa. 2015. Approximating Free-form Geometry with Height Fields for Manufacturing. In *Computer Graphics Forum*, Vol. 34. Wiley Online Library, 239–251.
- [14] Alexandra Ion, Johannes Frohnhofen, Ludwig Wall, Robert Kovacs, Mirela Alistar, Jack Lindsay, Pedro Lopes, Hsiang-Ting Chen, and Patrick Baudisch. 2016. Metamaterial Mechanisms. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16)*. ACM, New York, NY, USA, 529–539. DOI:<http://dx.doi.org/10.1145/2984511.2984540>
- [15] Luigi Malomo, Nico Pietroni, Bernd Bickel, and Paolo Cignoni. 2016. FlexMolds: Automatic Design of Flexible Shells for Molding. *ACM Trans. Graph.* 35, 6, Article 223 (Nov. 2016), 12 pages. DOI:<http://dx.doi.org/10.1145/2980179.2982397>
- [16] Jonàs Martínez, Jérémie Dumas, and Sylvain Lefebvre. 2016. Procedural Voronoi Foams for Additive Manufacturing. *ACM Trans. Graph.* 35, 4, Article 44 (July 2016), 12 pages. DOI:<http://dx.doi.org/10.1145/2897824.2925922>
- [17] James McCrae, Nobuyuki Umetani, and Karan Singh. 2014. FlatFitFab: interactive modeling with planar sections. In *UIST '14: Proceedings of the 27th annual ACM symposium on User interface software and technology*.
- [18] Robert Merz, FB Prinz, K Ramaswami, M Terk, and LE Weiss. 1994. Shape deposition manufacturing. In *1994 International Solid Freeform Fabrication Symposium*.
- [19] Stefanie Mueller, Bastian Kruck, and Patrick Baudisch. 2013. LaserOrigami: laser-cutting 3D objects. In *CHI '13: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*.
- [20] Stefanie Mueller, Tobias Mohr, Kerstin Guenther, Johannes Frohnhofen, and Patrick Baudisch. 2014. faBrickation: Fast 3D Printing of Functional Objects by Integrating Construction Kit Building Blocks. In *CHI'14: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM Press, New York, New York, USA, 3827–3834.



- [21] Steven Nagels, Raf Ramakers, Kris Luyten, and Wim Deferme. 2018. Silicone Devices: A Scalable DIY Approach for Fabricating Self-Contained Multi-Layered Soft Circuits Using Microfluidics. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 188, 13 pages. DOI:<http://dx.doi.org/10.1145/3173574.3173762>
- [22] Simon Olberding, Sergio Soto Ortega, Klaus Hildebrandt, and Jürgen Steimle. 2015. Foldio: Digital Fabrication of Interactive and Shape-Changing Objects With Foldable Printed Electronics. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. ACM, New York, NY, USA, 223–232. DOI:<http://dx.doi.org/10.1145/2807442.2807494>
- [23] Julian Panetta, Qingnan Zhou, Luigi Malomo, Nico Pietroni, Paolo Cignoni, and Denis Zorin. 2015. Elastic Textures for Additive Fabrication. *ACM Trans. Graph.* 34, 4, Article 135 (July 2015), 12 pages. DOI:<http://dx.doi.org/10.1145/2766937>
- [24] Raf Ramakers, Fraser Anderson, Tovi Grossman, and George Fitzmaurice. 2016. RetroFab: A Design Tool for Retrofitting Physical Interfaces Using Actuators, Sensors and 3D Printing. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 409–419. DOI:<http://dx.doi.org/10.1145/2858036.2858485>
- [25] Greg Saul, Manfred Lau, Jun Mitani, and Takeo Igarashi. 2011. SketchChair: an all-in-one chair design system for end users. In *TEI '11: Proceedings of the fifth international conference on Tangible, embedded, and embodied interaction*. New York, New York, USA, 73–80.
- [26] Valkyrie Savage, Ryan Schmidt, Tovi Grossman, George Fitzmaurice, and Björn Hartmann. 2014. A Series of Tubes: Adding Interactivity to 3D Prints Using Internal Pipes. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology (UIST '14)*. ACM, New York, NY, USA, 3–12. DOI:<http://dx.doi.org/10.1145/2642918.2647374>
- [27] Ryan Schmidt and Karan Singh. 2010. Meshmixer: an interface for rapid mesh composition. In *ACM SIGGRAPH 2010 Talks*. ACM, 6.
- [28] Martin Schmitz, Mohammadreza Khalilbeigi, Matthias Balwierz, Roman Lissermann, Max Mühlhäuser, and Jürgen Steimle. 2015. Capricate: A Fabrication Pipeline to Design and 3D Print Capacitive Touch Sensors for Interactive Objects. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. ACM, New York, NY, USA, 253–258. DOI:<http://dx.doi.org/10.1145/2807442.2807503>
- [29] Christian Schumacher, Bernd Bickel, Jan Rys, Steve Marschner, Chiara Daraio, and Markus Gross. 2015. Microstructures to Control Elasticity in 3D Printing. *ACM Trans. Graph.* 34, 4, Article 136 (July 2015), 13 pages. DOI:<http://dx.doi.org/10.1145/2766926>

- [30] Rupinder Singh. 2010. Three dimensional printing for casting applications: A state of art review and future perspectives. In *Advanced Materials Research*, Vol. 83. Trans Tech Publ, 342–349.
- [31] Pitchaya Sitthi-amorn, Javier E Ramos, Yuwang Wangy, Joyce Kwan, Justin Lan, Wenshou Wang, and Wojciech Matusik. 2015. MultiFab: a machine vision assisted platform for multi-material 3D printing. *ACM Transactions on Graphics (TOG)* 34, 4 (July 2015), 129–129:11.
- [32] P S Sreetharan, J P Whitney, M D Strauss, and R J Wood. 2012. Monolithic fabrication of millimeter-scale machines. *Journal of Micromechanics and Microengineering* 22, 5 (May 2012), 055027.
- [33] Udayan Umapathi, Hsiang-Ting Chen, Stefanie Mueller, Ludwig Wall, Anna Seufert, and Patrick Baudisch. 2015. LaserStacker: Fabricating 3D Objects by Laser Cutting and Welding. In *UIST '15: Proceedings of the 28th annual ACM symposium on User interface software and technology*. ACM Press, New York, New York, USA, 575–582.
- [34] Nobuyuki Umetani and Ryan Schmidt. 2017. SurfCuit: Surface-Mounted Circuits on 3D Prints. *IEEE computer graphics and applications* 37, 3 (2017), 52–60.
- [35] Kiril Vidimce, Alexandre Kaspar, Ye Wang, and Wojciech Matusik. 2016. Foundry: Hierarchical Material Design for Multi-Material Fabrication. In *UIST '16: Proceedings of the 29th annual ACM symposium on User interface software and technology*. ACM, New York, New York, USA, 563–574.
- [36] Kiril Vidimce, Szu-Po Wang, Jonathan Ragan-Kelley, and Wojciech Matusik. 2013. OpenFab: a programmable pipeline for multi-material fabrication. *ACM Transactions on Graphics (TOG)* 32, 4 (July 2013), 136.
- [37] Guanyun Wang, Lining Yao, Wen Wang, Jifei Ou, Chin-Yi Cheng, and Hiroshi Ishii. 2016. xPrint: A Modularized Liquid Printer for Smart Materials Deposition. In *CHI '16: Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, New York, New York, USA, 5743–5752.
- [38] Christian Weichel, Manfred Lau, and Hans Gellersen. 2013. Enclosed: A Component-centric Interface for Designing Prototype Enclosures. In *Proceedings of the 7th International Conference on Tangible, Embedded and Embodied Interaction (TEI '13)*. ACM, New York, NY, USA, 215–218. DOI : <http://dx.doi.org/10.1145/2460625.2460659>
- [39] Karl Willis, Eric Brockmeyer, Scott E Hudson, and Ivan Poupyrev. 2012. Printed Optics: 3D Printing of Embedded Optical Elements for Interactive Devices. In *UIST '12: Proceedings of the 25th annual ACM symposium on User interface software and technology*. ACM Press, New York, New York, USA, 589–598.
- [40] Junichi Yamaoka, Ryuma Niiyama, and Yasuaki Kakehi. 2017. BlowFab: Rapid Prototyping for Rigid and Reusable Objects using Inflation of Laser-cut Surfaces. In

- Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*. ACM, New York, New York, USA, 461–469.
- [41] Jonas Zehnder, Espen Knoop, Moritz Bächer, and Bernhard Thomaszewski. 2017. Metasilicone: design and fabrication of composite silicone with desired mechanical properties. *ACM Transactions on Graphics (TOG)* 36, 6 (Nov. 2017), 240–13.