STEFANIE MUELLER

CONSTRUCTABLE AND LASERORIGAMI: INTERACTIVE CONSTRUCTION OF FUNCTIONAL TWO- AND THREE-DIMENSIONAL PHYSICAL OBJECTS USING A LASER-CUTTER.

CONSTRUCTABLE AND LASERORIGAMI: INTERACTIVE CONSTRUCTION OF FUNCTIONAL TWO- AND THREE-DIMENSIONAL PHYSICAL OBJECTS USING A LASER-CUTTER.

STEFANIE MUELLER

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science in IT Systems Engineering

Human-Computer Interaction Group
Hasso Plattner Institute
University of Potsdam

October 2013

Stefanie Mueller: *constructable and LaserOrigami: Interactive Construction of Functional Two- and Three-Dimensional Physical Objects Using a Laser-Cutter.* October 2013

SUPERVISOR:
Prof. Dr. Patrick Baudisch

To Mom and Dad

For your endless support and encouragement during my studies.

ABSTRACT

Personal fabrication tools, such as 3D printers, milling machines, and laser cutters, allow users to prototype physical objects. During the last years, the acquisition costs of these devices drastically decreased. Still, only few people own a personal fabrication device yet.

One reason is the current workflow that does not support a good learning curve due to the lack of immediate feedback. Users need to plan everything ahead in a 3D editor, and only after all editing is done can fabricate and test the physical object. In the case of 3D printers, every iteration can take hours or even needs overnight printing.

While the problem of lacking feedback due to slow hardware is new in personal fabrication, we have seen a similar development in the past with personal computing. On early computers, processing the input took hours or even days to finish. Only when faster computing hardware was developed, a change in the interaction paradigm towards interactive computing became possible, which ultimately led to the wide spread of personal computers. In interactive computing, every input results in immediate feedback.

Researchers argue that bringing the qualities of interactive computing to personal fabrication will make personal fabrication devices more accessible and thus will ultimately lead to a wider spread. In the related work, this concept of interactive editing in personal fabrication is called *interactive fabrication*. The idea is to let users once again work directly on the workpiece and to create physical output after every editing step, as known from traditional crafting.

The transition to interactive fabrication, however, comes at a cost. First, since editing is now intertwined with fabrication, editing becomes slower, because users have to repeatedly wait for the fabrication engine to finish. Second, as we give up on traditional tools like CAD, users lose the precision required to create functional devices.

In this thesis, we attempt to put these qualities back into interactive fabrication, moving it in the direction of what we call *interactive construction*. We present constructable, a laser cutter-based interactive construction system that allows users to construct functional mechanical devices. In constructable, users interact by drafting directly on the workpiece using a hand-held laser pointer. The system tracks the pointer, beautifies its path, and implements its effect by

cutting the workpiece using a fast high-powered laser cutter. constructable achieves precision through tool-specific constraints, user-defined sketch lines, and by using the laser cutter itself for all visual feedback, rather than using a screen or projection.

To allow for interactive construction of three-dimensional objects, we developed a novel fabrication technique called LaserOrigami and integrated it into constructable. Since 3D printers are too slow for interactive construction, LaserOrigami uses a fast laser cutter to construct 3D objects by folding and stretching the workpiece. LaserOrigami achieves this by heating up selected regions of the workpiece until they become compliant and bend down under the forces of gravity. LaserOrigami administers the heat by defocusing the laser, which distributes the laser's power across a larger surface.

ZUSAMMENFASSUNG

Personal-Fabrication-Geräte, wie Laserschneider und 3D Drucker, ermöglichen es Benutzern präzise Objekte schnell herzustellen. Die Benutzer arbeiten dabei jedoch an einem CAD System und sind vom eigentlichen Werkstück entfernt. Aktuelle Geräte zum interaktiven Fabrizieren ermöglichen das direkte Arbeiten auf dem Werkstück wieder, allerdings sind sie nicht präzise.

Unser System Constructable ist ein interaktiver Zeichentisch, der auf einem Laserschneider basiert und physische Ausgaben nach jedem Interaktionsschritt erzeugt. Benutzer interagieren indem sie mit einem Laserpointer direkt auf das Werkstück zeichnen. Constructable verfolgt den Laserpointer, richtet den Pfad korrekt aus und schneidet das Werkstück mit dem Laserschneider.

Constructable erreicht Präzision durch Laserpointer-spezifische Funktionalität, benutzerdefinierte Ausrichtungslinien, und die Verwendung des Laserschneiders für die gesamte Ausgabe anstelle eines Bildschirms oder einer Projektion. Wir zeigen wie Constructable es erlaubt einfache funktionale Geräte herzustellen, die nicht mit bisherigen interaktiven Fabrikationsansätzen hergestellt werden können, wie zum Beispiel ein Fahrgetriebe.

Da Constructable auf einem Laserschneider basiert, können Benutzer nur zwei-dimensionale Teile ausschneiden, die am Ende vom Benutzer zu einer drei-dimensionalen Struktur zusammengebaut werden müssen. LaserOrigami löst das Problem, indem es den Laserschneider nicht nur zum Schneiden benutzt sondern auch zum automatischen Falten des Werkstücks, sodass der manuelle Zusammenbau nicht mehr nötig ist.

LaserOrigami heizt ausgewählte Bereiche des Werkstücks auf bis sie elastisch werden und sich aufgrund der Schwerkraft nach unten biegen. LaserOrigami regelt die Wärme indem es den Laser defokussiert, sodass sich die Energie des Lasers auf einen größeren Bereich verteilt. LaserOrigami bündelt Schneiden und Biegen in einen integrierten Prozess, indem es den Schneidetisch automatisiert nach oben und unten bewegt. LaserOrigami hat drei Gestaltungselemente: den Knick, den Hänger, und die Dehnung. Wir zeigen wie diese drei Gestaltungselement es erlauben eine Vielzahl verschiedener Objekte herzustellen.

PUBLICATIONS

This thesis is based on the following two publications:

Mueller, S., Lopes, P., Baudisch, P. Interactive Construction: Interactive Fabrication of Functional Mechanical Devices. In *Proceedings of UIST'12*, pp. 599-606. (Fullpaper, 8 pages) [39]

Mueller, S., Kruck, B., Baudisch, P. LaserOrigami: Laser-Cutting 3D Objects. In *Proceedings of CHI'13*, pp. 2585-2592. (Fullpaper, 8 pages) **[Best Paper Award]** [38]

ACKNOWLEDGMENTS

First of all, I would like to express my sincere gratitude to my advisor Prof. Dr. Patrick Baudisch for the continuous support of my research, for his motivation, enthusiasm, and immense knowledge. His guidance allowed me to gain important insights in what constitutes excellent research. I could not have imagined having a better advisor for my studies.

Besides my advisor, I would like to thank the members of the HCI lab: Dr. Anne Roudaut, Dr. Dominik Schmidt, Christian Holz, Sean Gustafson, Pedro Lopes, and Lung-Pan Cheng for the stimulating discussions and for the sleepless nights we were working together before deadlines.

Special thanks to Konstantin Kaefer, Bastian Kruck, and Tobias Mohr for their support during the TEI'13 and CHI'13 conference demonstrations of constructable and LaserOrigami. Without your help these demos would not have been as successful.

I also want to thank Mike Sinclair for sharing his wisdom on uncommon laser-cutting techniques.

Finally, I would like to thank my family: my parents Juergen Mueller and Siglinde Mueller, for supporting me throughout all my studies, my sister Kristin Mueller for always having a sympathetic ear, and my boyfriend Philipp Berger for all his encouragement during the last two years.

CONTENTS

LIST OF FIGURES

1

INTRODUCTION

Personal fabrication tools, such as 3D printers, milling machines, and laser cutters, allow users to prototype physical objects quickly. They allow users to create decorative objects, such as a set of custom chess figures, as well as mechanical objects, such as a clock with all its gears and mounts.

During the last years, personal fabrication tools became more widely available as the acquisition costs drastically decreased. While professional 3D printers are still expensive (e.g. 90.000 Euro for our Dimension sst 1200es [1] 3D printer), a MakerBot [36] 3D printer costs only around 2000 Euro, while still having sufficient shape resolution for fabrication projects that even involve mechanical parts such as gears. Similar developments can be seen for laser cutters (e.g. 45.000 Euro for the PLS6.150D from Universal Laser Systems [28] vs. 1000 Euro for the LS3020 from HPC Laser LTD [27]), and milling machines (e.g. ShapeOko [52] milling machine for 450 Euro).

While one might expect that the decreased price leads to a high number of sold machines, actually very few people own a personal fabrication device yet (e.g., in the US 23.000 devices were sold in 2011 [62]). People who actually get in touch with personal fabrication, do this mostly through community-driven fabrication spaces (so called FabLabs [22]), which offer fabrication equipment to the public. These FabLabs are lead by a small number of expert users that guide beginners through the fabrication process.

These expert users are required, since the current fabrication workflow does not support a good learning curve, but begins with a lot of frustration. Users need to plan everything ahead in a complex dig-

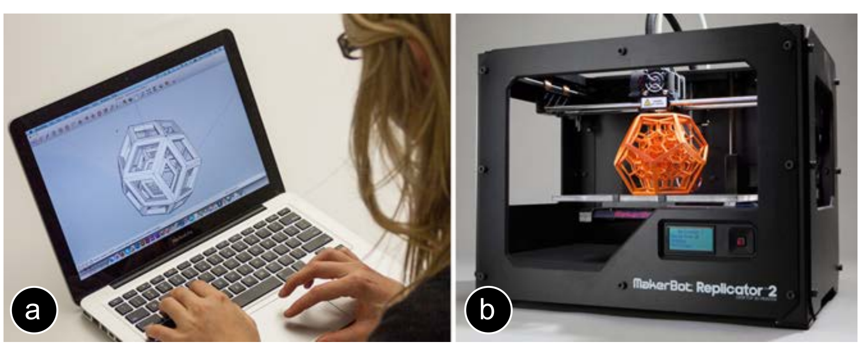

Figure 1: The current workflow requires users to plan everything ahead in a 3D editor. The fabrication of the object takes place only after all editing is done. If users make mistakes, they have to reprint the entire model.

ital modeling tool, and only after all editing is done can fabricate and test the actual physical object. If they made mistakes, they have to reprint the entire model. In the case of 3D printers this can take hours or even needs overnight printing. This becomes especially frustrating, as many users lack the modeling skills to create the 3D model. Even 2D tools, such as laser cutters, require expert knowledge when creating the 2D layout for the cut path, as the 2D parts need to be assembled later on using joints that need to match perfectly.

Figure 2: 3D Printer advertisement in a german newspaper with the headline: everyone sleeps, only one works. Making fun of the slow fabrication speed of 3D printers.

While the problem of lacking feedback due to slow hardware is new in personal fabrication, we have seen a similar development in the past with personal computing. The first computers were slow and did not allow for immediate feedback as processing the input took hours or even days to finish. Users had to hand in their punched cards in the evening, had to wait overnight, and were only able to check the results in the next morning. If there was a mistake, they had to create a new punched card to correct it, and wait another night, just to see the next day that there was another problem. As a result, many people shied away from using computers. In addition,

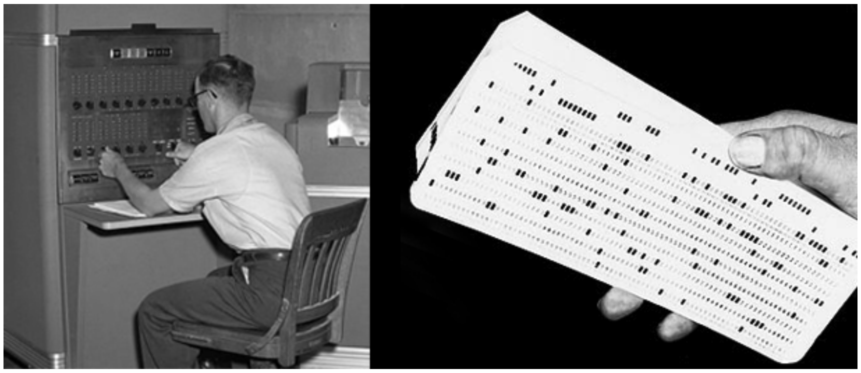

Figure 3: Processing such a punched card in early computing took hours and was often done overnight due to slow hardware.

punched cards represented a very abstract representation of the data that was hard to access and hard to learn by new users.

Only when faster computing hardware became available, a change in the interaction paradigm became possible, which ultimately led to the wide spread of personal computers today. This change was introduced by Doug Engelbart who presented the new interactive computing paradigm in 1968 [16]. He used a computer monitor with a graphical user interface and a pointing device, the mouse, to work directly on the data displayed on the screen. For each interaction, he could see the result immediately, i.e. a file that was clicked was marked as selected, a double clicked folder opened and showed the contained files. Users were now able to work directly on the data and to get immediate feedback, which facilitated the usage of personal computers tremendously. Soon, many people got one at home.

Figure 4: Faster hardware allowed for interactive computing that provided immediate feedback after every input.

Researchers argue that bringing the qualities of interactive computing to personal fabrication, will make personal fabrication devices more accessible and thus will ultimately lead to a wider spread of personal fabrication. In personal fabrication, directly working on the data essentially means to work directly on the workpiece as we know it from traditional crafting. Generating an input and changing the data, is equivalent to applying a tool to the workpiece and changing its shape or texture. Getting immediate feedback is thus the immediate physical change of the workpiece. Figure 5 illustrates this: Here the craftsman uses a chisel to carve a piece of wood. With every input he creates with his chisel, he immediately sees the physical output, i.e. the changes of the physical matter. This process is very easy to access as even kids now how to shape an object with basic tools and their hands.

Figure 5: Immediate feedback in personal fabrication means to see the workpiece change after every editing step, as in traditional crafting.

In the related work, this concept of interactive editing in personal fabrication is called *interactive fabrication* [61]. Shaper [61], for example, allows users to create foam drops from a foam-printer using a co-located transparent touchscreen (Figure 6). The immediate physical output allows users to (1) validate their designs early and (2) build subsequent work steps on the result of earlier steps. The related work suggests that this offers value to artists and designers [61], as their creative process is often inspired by seeing the partially completed workpiece. We argue that similar benefits can be achieved when working on technical projects with functional objects.

The transition to interactive fabrication, however, comes at a cost. First, since editing is now intertwined with fabrication, editing becomes slower, because users have to repeatedly wait for the fabrication engine to finish. Second, as we give up on traditional tools like CAD, users lose the precision required to create functional devices.

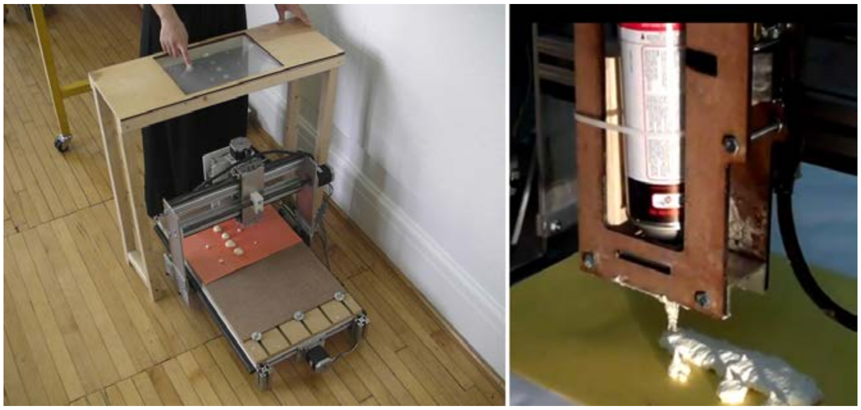

Figure 6: Previous work used interactive editing in personal fabrication only to create artistic shapes, as all of the input is created freehand and thus does not provide the precision required for creating functional objects.

In this thesis, we attempt to put the qualities of CAD back into interactive fabrication, moving it in the direction of what we call *interactive construction*. We present constructable, a laser cutter-based interactive construction system that allows users to construct functional mechanical devices, while maintaining the immediateness of an interactive fabrication system. In constructable, users interact by drafting directly on the workpiece using a hand-held laser pointer. The system tracks the pointer, beautifies its path, and implements its effect by cutting the workpiece using a fast high-powered laser cutter. constructable achieves precision through tool-specific constraints, user-defined sketch lines, and by using the laser cutter itself for all visual feedback, rather than using a screen or projection. In chapter 3, we demonstrate how constructable allows creating simple but functional devices, including a simple gearbox, that cannot be created with traditional interactive fabrication tools.

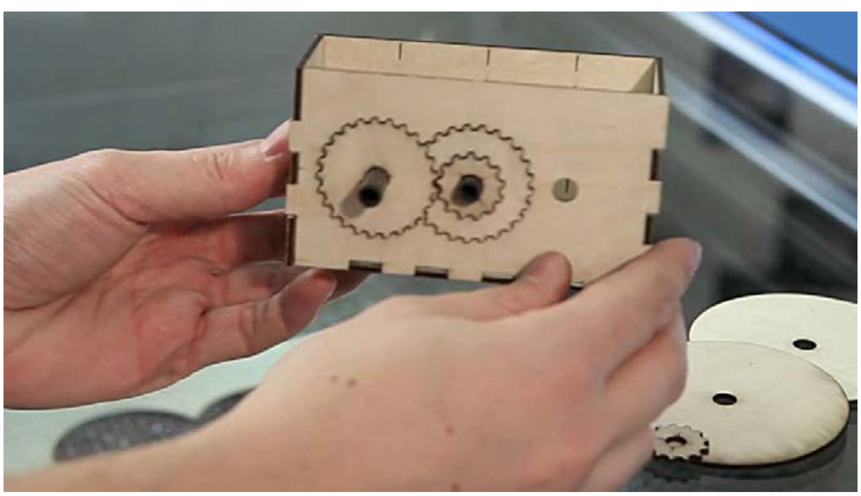

Figure 7: Unlike previous work, our system constructable allows users to interactively fabricate functional mechanical objects, such as this gearbox.

constructable shows how to interactively create two-dimensional functional objects using a laser-cutter. To allow for interactive creation of 3D objects, we further developed a rapid-prototyping system called LaserOrigami that uses a novel laser-bending technique to produce 3D objects. LaserOrigami is substantially faster than traditional 3D fabrication techniques such as 3D printing and unlike traditional laser cutting the resulting 3D objects require no manual assembly. The key idea behind LaserOrigami is that it achieves three-dimensionality by folding and stretching the workpiece, rather than by placing joints, thereby eliminating the need for manual assembly. LaserOrigami achie- ves this by heating up selected regions of the workpiece until they become compliant and bend down under the forces of gravity. LaserOrigami administers the heat by defocusing the laser, which distributes the laser's power across a larger surface. LaserOrigami implements cutting and bending in a single integrated process by automatically moving the cutting table up and down—when users take out the workpiece, it is already fully assembled.

Figure 8: We allow for interactive construction of three-dimensional objects with our system LaserOrigami. LaserOrigami uses a fast high-powered laser cutter to create 3D objects through cutting and bending.

2

RELATED WORK

The work in this thesis builds on previous research in personal fabrication, digital modeling for personal fabrication, and interactive fabrication.

2.1 PERSONAL FABRICATION

With personal fabrication tools researchers commonly refer to some superset of milling machines, laser cutters, and 3D printing machines. The main goal of personal fabrication is to allow for rapid prototyping, with the intent of reduced costs compared to actual fabrication, thereby allowing for a more iterative design process.

The concept has support not only in research, but also in the open source/DIY community, which is leading the development of more accessible machines, such as *Lasersaur* [29] for laser-cutters, *MakerBot* [36] for 3D printers, and *Shapeoko* [52] for milling machines (Figure 9).

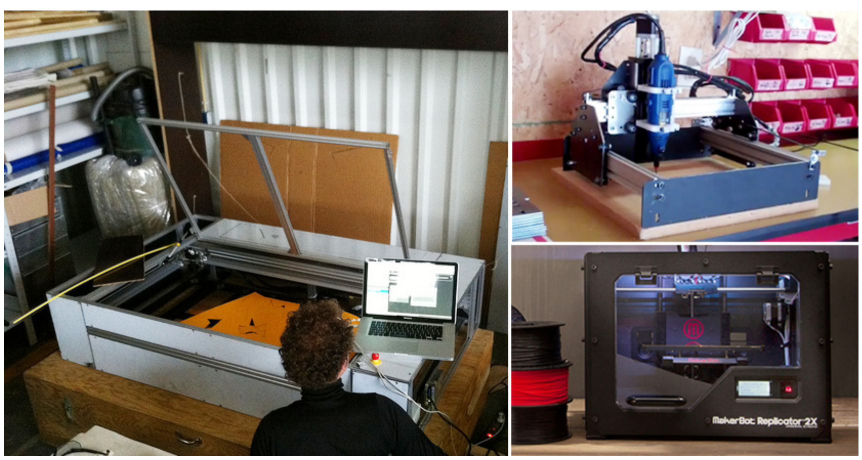

Figure 9: open source versions of the most common personal fabrication tools: (a) *Lasersaur* laser-cutter, (b) *Shapeoko* milling machine, (c) *MakerBot* 3D printer.

The fabrication of one-off objects allows for many new application areas. For instance, 3D printed information visualizations can communicate spatial information more effectively than digital representations [25]. On the flipside, *Shadowgram* [41] uses 3D printed objects for social brainstorming to create group activities around them. Fi-

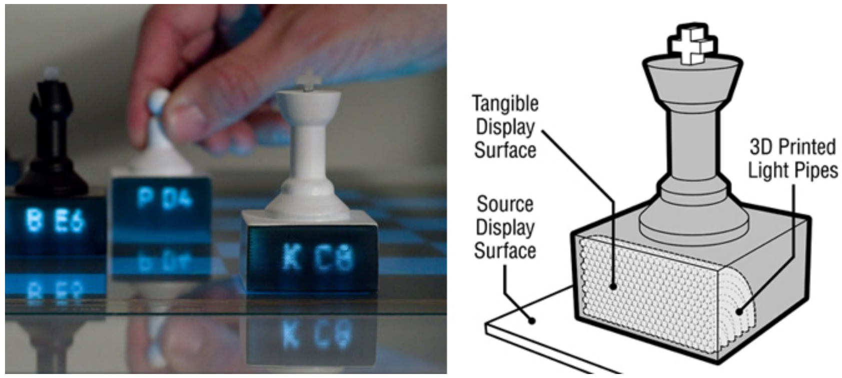

Figure 10: *Printed Optics* creates this display element by using optical clear material for 3D printing. This allows to print glass fibers inside the object.

nally, *Viztouch* [10] shows how 3D printed objects can be created for blind students that need tactile feedback to perceive information. In addition, Yahoo Research Japan recently presented a voice-activated search engine that 3D prints objects as the result of a search [63].

Beside the fabrication of passive objects, researchers investigate how to create interactive prototypes using current fabrication technology. Figure 10 shows how *Printed Optics* [59] uses optical clear material in 3d printing to fabricate optical sensors and display elements for interactive devices. *Papillon* [9] extends the work on printed optics for curved display surfaces. *Midas* [51] on the other hand uses a vinyl cutter and copper sheets to fabricate capacitive touch prototypes (Figure 11). *Sauron* [50] finally makes 3d printed models interactive by tracking moving components with a single attached camera.

Many workshops equipped with rapid-prototyping tools (denoted as *fab labs* [22]) now offer their services to a broader audience, leading to an increased interest in personal fabrication. This so called *democratization of technology*, i.e. that the creation of high-fidelity physical prototypes is no longer restricted to professional designers, creates a

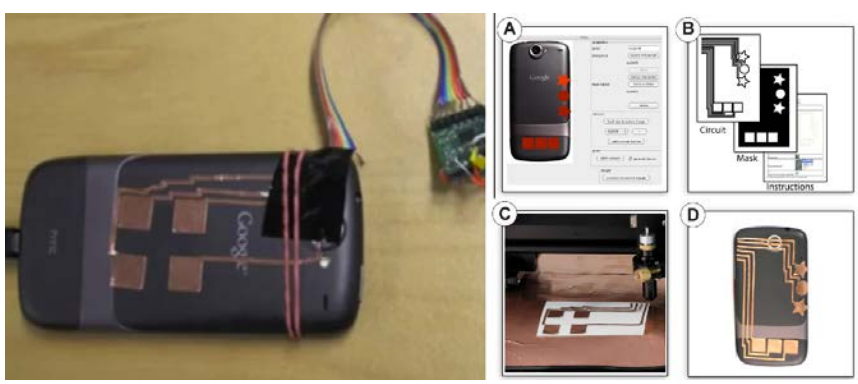

Figure 11: *Midas* allows users to create interactive prototypes with capacitive touch sensing quickly. Users layout the touch elements in the Midas software, then fabricate them on a vinyl cutter.

large potential for new innovations [55]. Researchers investigate how even young children can get engaged with these new creative technologies [44] that are currently difficult to access for novices.

2.2 DIGITAL MODELING FOR PERSONAL FABRICATION

The traditional workflow of personal fabrication tools is that users sit in front of a computer and model the 2D or 3D object in a Computer Aided Design Software (CAD), such as OpenDraw, Illustrator or GoogleSketchup. After they are done with editing their model, they send it to the personal fabrication device via the regular printing interface.

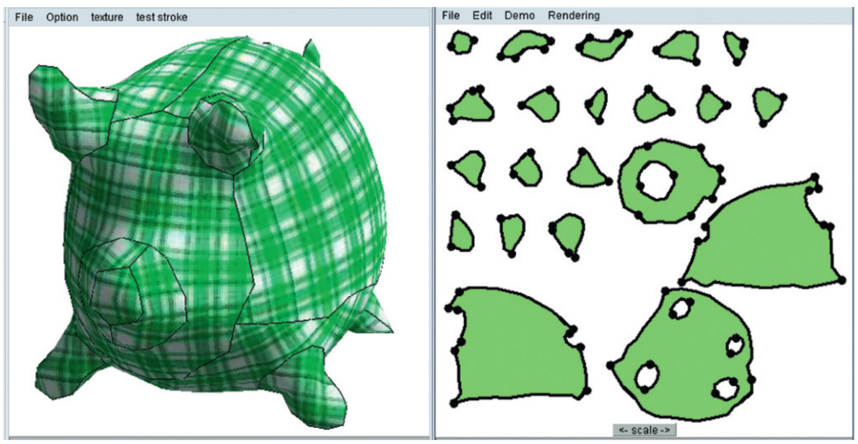

Figure 12: *Plushie* is a 3D editor for creating plush toys. Besides the perspective view, Plushie displays the 2D parts for sewing.

Since modeling the physical object in the digital realm is a complex task, a range of projects aim at lowering the entry barrier to 2D and 3D modeling by restricting the space of possible objects. For instance, *SketchChair* [48] provides a simple editor for sketching chair designs. The editor also supports an avatar of the user's body to test if the chair fits the user's body proportions. At the end, the software decomposes the chair drawing into a set of laser-cuttable pieces with matching joints. *FurnitureFactory* [42] uses a similar approach, but lets users draw a perspective sketch of the desired furniture on a piece of paper and then decomposes the 3d drawing into 2D pieces. Another example is *Plushie* [37] shown in Figure 12, which supports the user in modeling plush animals by providing a set of constrained interactions. *Plushie* also provides a second view that shows the 2D pieces that the user sews together at the end. Finally, tools like Lau's *modeling in context* [30] allow users to draw a three-dimensional shape on top of a photo to make it fit existing objects in the real world.

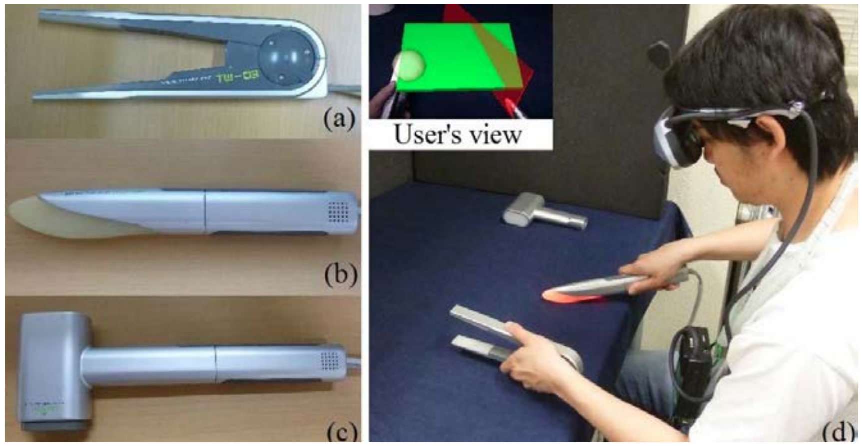

Figure 13: *Virtual Tooling* uses real tools for 3D editing. For instance, the knife cuts an object into two pieces.

Other projects use embodied spatial input for 3d modeling. *Virtual Tooling* [4], for instance, is a mixed reality handcrafting system that uses real tools, such as a hammer, knife, and tweezer to cut, join, and assemble digital shapes (Figure 13). The system also provides a physical ruler that works as both a measurement device and an alignment aid. The physical devices are tracked and registered with the virtual environment.

Dress-up [58] uses a similar approach for clothing design: The user uses handheld tools to draw the shape of the dress around a physical mannequin (Figure 14). The digital representation of the dress is displayed on a separate monitor. As in *Virtual Tooling* users have different devices to achieve different functionality, such as a cutting tool and a surface tool. Finally, *SpatialSketch* [60] uses mid-air hand-gestures to define the shape of the geometry. Users draw with an infrared pen light into mid-air and by pressing a button turn the recording on and off. The location of the pen light is tracked using two infrared cam-

Figure 14: *Dress-up* allows users to draw the shape of a dress around a physical mannequin using tangible tools.

eras for stereo vision 3D input. As with the previously mentioned systems, physical output is only provided at the end of the design process.

2.3 INTERACTIVE FABRICATION

In the traditional fabrication workflow, users sit at the computer, edit the digital model and only after all editing is done, the actual fabrication of the physical object starts. In contrast, *interactive fabrication systems* [61] do not require the user to model the entire object before fabrication, but offer an alternation between user and system control. Thereby, interactive fabrication systems offer a tighter feedback loop since physical output after every editing step provides feedback earlier on and allows users to make changes even within a version. Since users need to wait for the machine after each input, fast fabrication is a key element of interactive fabrication systems to keep users in the flow [14].

One of the first interactive fabrication systems is *ModelCraft* [53], which allows users to modify physical paper models by annotating directly onto the model using an AnotoPen (Figure 15). *ModelCraft* then implements the changes by reprinting the model, which users have to manually fold into shape.

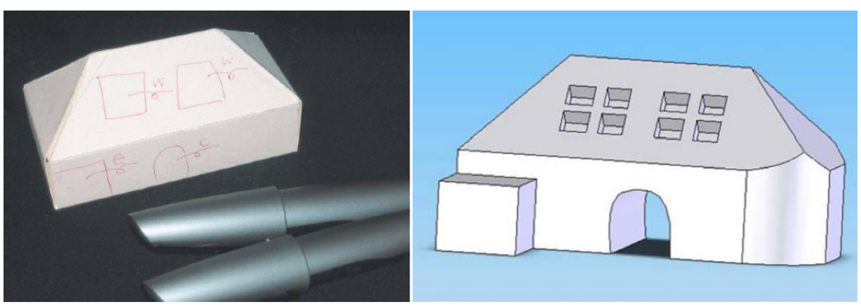

Figure 15: In *Modelcraft* users annotate directly on the paper model, the changes are applied to the 3D model, and a new paper model is printed.

The first interactive fabrication system that used a personal fabrication device to speed-up the feedback loop and to remove manual labor is *CopyCAD* [19]. Figure 16 shows how the *CopyCAD* system allows users to control a milling machine by either drawing directly onto the piece of wood to create new geometry from scratch or by copying the shape of an existing physical object. The system uses an overhead digital camera to capture physical objects, a projector to display captured geometry, and a line-laser for multitouch input. The idea of applying operations to the object based on its shape was first proposed by *LaserCooking* [20], which identified specific food items in the laser-cutter and handled them appropriately.

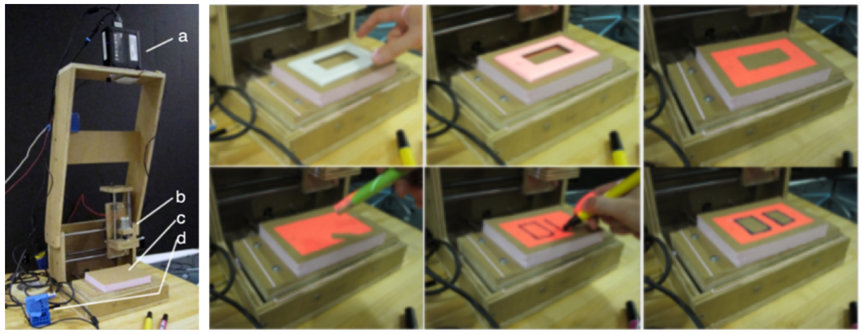

Figure 16: *CopyCAD* allows users to copy and remix geometry from physical objects by using a camera on top of a milling machine.

Additive fabrication devices, such as 3D printers, are difficult to use for interactive fabrication purposes, since the voxel-wise creation of material is very slow. *Shaper* [61]) therefore uses a foam printer with expanding foam, which speeds up the fabrication process but at the cost of precision (Figure 17). *Shaper's* foam printer uses a co-located transparent touchscreen for interaction: when users touch the screen at a certain position, a single foam drop is fabricated at the corresponding location in the work area.

Figure 17: *Shaper* consists of a transparent touchscreen co-located with a foam-printer for the physical output.

Since immediate physical output does not allow for an undo-function as known from software, a group of projects tries to minimize error by using a virtual model as reference for fabrication. During fabrication, users can decide if they want to stick to the digital model or if they want to locally overwrite it with a different shape. *FreeD* [64], for instance, is a handheld milling machine that slows down its spindle when the user approaches the virtual model that was previously aligned with the block of material (Figure 18). Users can use the virtual model as a rough shape baseline, and then use the overwrite-button to customize the design. A second paper on the *FreeD* provides additional concepts for this new kind of hybrid carving [65],

Figure 18: *FreeD* is a milling machine that prevents users from making mistakes. Users specify a model beforehand but can overwrite it on the fly.

such as remixing several virtual models during fabrication or browsing through a database of shapes during fabrication.

Figure 19 shows *HapticIntelligentsia* [31], a similar project for additive fabrication that uses a glue gun attached to a phantom device. Users first load a virtual model into the system and then receive force-feedback when approaching the model location in space. This guidance allows them to fabricate an object with little to no previous experience. *Position Correcting Tools* [47] shows the same concept for a 2D router.

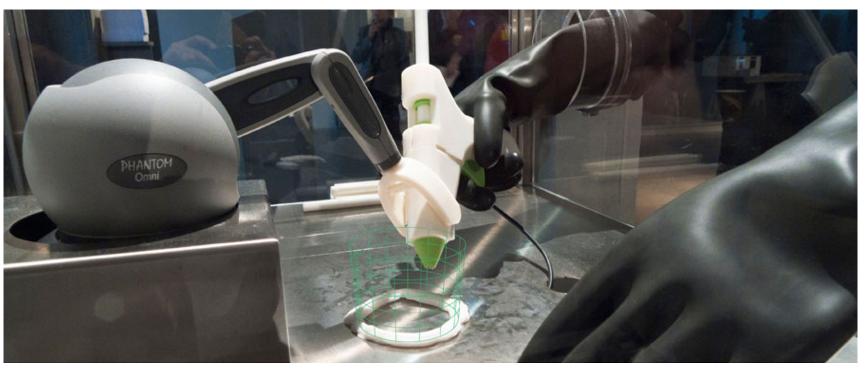

Figure 19: *HapticIntelligentsia* uses a glue gun attached to a phantom device to guide users during fabrication using force feedback.

With our system *constructable*, we extend interactive fabrication towards precision without requiring a predefined virtual model. Our *interactive construction* approach uses a set of tool-based constraints, which allows the user to even fabricate mechanical devices that require all parts to match and work together to function properly.

3 CONSTRUCTABLE: INTERACTIVE LASERCUTTING

constructable is a drafting table that produces physical output in every step. As illustrated by Figure 20, all interaction in constructable takes place on the workpiece, mediated through low-power hand-held laser pointers, which we call proxy lasers or simply tools. In the shown example, the user uses the finger joint tool to add finger joints between two pieces by crossing the two involved edges.

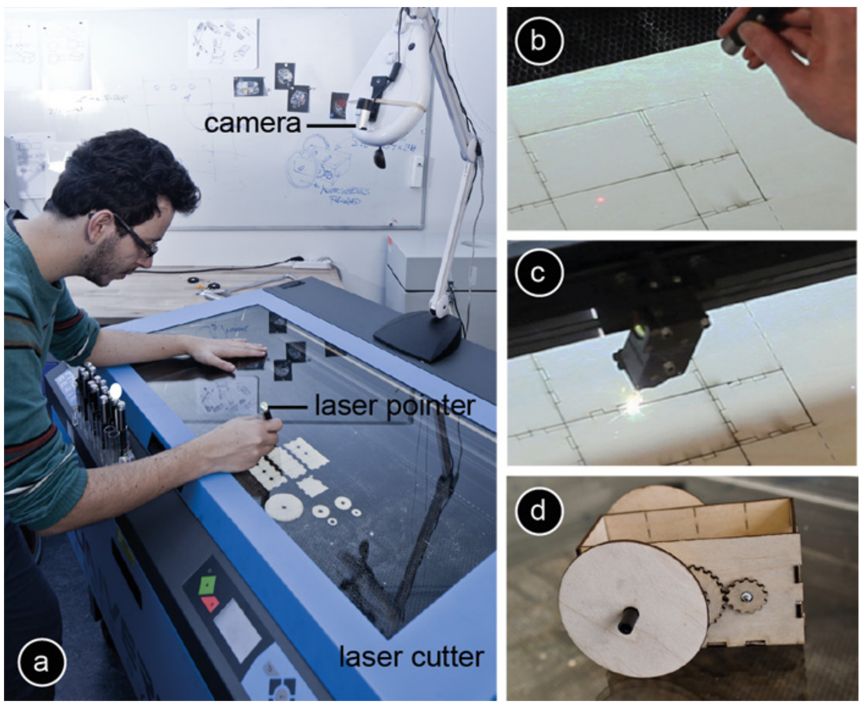

Figure 20: (a) constructable users interact by drafting directly on the workpiece with hand-held lasers. (b) Here the user sketches a finger joint across two objects (c) The system responds by cutting the desired joint using the cutting laser. (d) constructable allows creating precise functional mechanical objects, such as this simple motorized vehicle.

Proxy lasers are too weak to affect the work piece. To make the interaction 'real', constructable tracks proxy laser interactions using a camera mounted above (Figure 20a), reconstructs the tool's path, transforms it using a constraint set defined by the current tool, and implements the effect using its high-powered cutting laser (Figure 20c). Since all key elements were constructed in the context of constraints, we obtain fully functional mechanical devices, such as the simple motorized vehicle shown in Figure 20d.

The combination of proxy laser and cutting laser is one of the key ideas behind our system, because it allows users to work faster and more interactively than the interactive fabrication systems in the related work. The reason is that the manipulation of materials that are durable enough for making functional mechanical devices require high-powered tools. These, however, require safety measures, such as the laser cutter's glass enclosure. Opening and closing the enclosure for each interaction takes too long, but by letting users point at the workpiece through the safety enclosure, constructable allows user's attention to remain on the workpiece at all times, despite the enclosure. As intended, users witness how the workpiece changes with every interaction first hand and not mediated by a screen or projector.

constructable achieves its other main design objective, i.e., precision, using a system of constraints implemented by the individual proxy lasers (Figure 21).

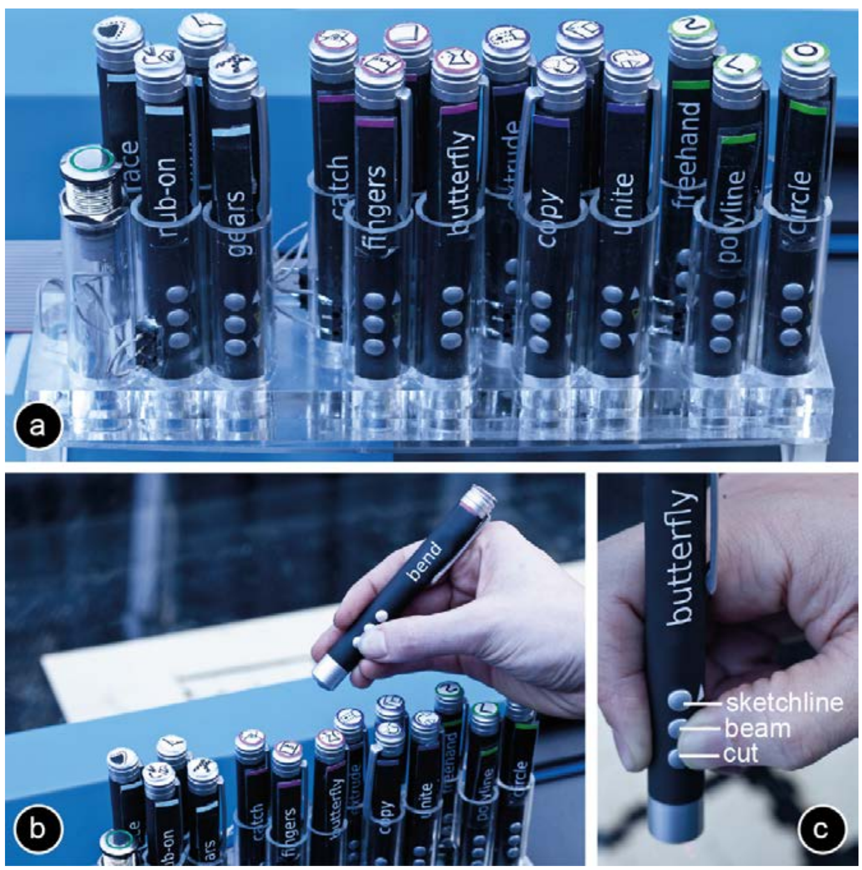

Figure 21: (a, b) constructable offers 15 different proxy lasers and an undo button. (c) Each proxy-laser offers three barrel buttons.

3.1 PROXY LASERS

Figure 21 provides a closer look at the proxy lasers. Each proxy laser features three barrel buttons (Figure 21c). While held depressed, the middle button activates the beam, allowing the system and the user to see where the tool is pointed [40]. The visual feedback allows users to determine a starting point with precision before starting to cut. It thereby implements the tracking state of its three-state model [12].

The other two buttons trigger the tool's two modes of operation. The cut button allows cutting a tool-specific shape, such as a circle for the circle tool. The sketch line button creates the same shape, but etches it as a shallow dashed line into the surface of the material. Sketch lines have no direct impact on the mechanics of the workpiece, but instead serve as alignment aids that magnetically attract subsequent cuts (alignment lines [8]).

All tools explain themselves exclusively through the cut or sketch line they produce and there is no further visual feedback, i.e., no screen or projector. Since all output is created using the laser cutter itself, constructable's geometry output is extremely precise.

3.2 CREATING, SELECTING, COPYING, AND PASTING USING TOOLS

constructable achieves precision by means of sketch lines and by implementing constraints into every proxy laser. Constraints differ between tools.

Creating: *Polyline, circle,* and *freehand* are constructable's tools for creating objects from scratch. These tools are only moderately constrained. The *circle* tool, for example, always produces a perfect circle, but diameter and location remain freehand. The *freehand* tool is not subject to any constraints.

Most of *constructable's* tools connect to or extend an existing object and this spatial relationship adds constraints. Users establish these constraints by selecting one or more existing objects. The *finger joint* tool, for example, snaps to existing lines.

Selecting: As illustrated by Figure 22, users select (a) a surface by clicking into it, (b) an edge by crossing it (see *Crossy* [3]), and (c) a point by drawing a pigtail close to it (see *Scriboli* [23]). We designed this selection mechanism so as to extend seamlessly to multiple objects. Users select (d) multiple surfaces by drawing a path across, (e) multiple edges by crossing multiple edges, and (f) multiple points as a sequence of multiple pigtails.

Pasting: A range of tools, such as the *copy* tool, results in the creation of new objects. The size and shape of a new object is determined

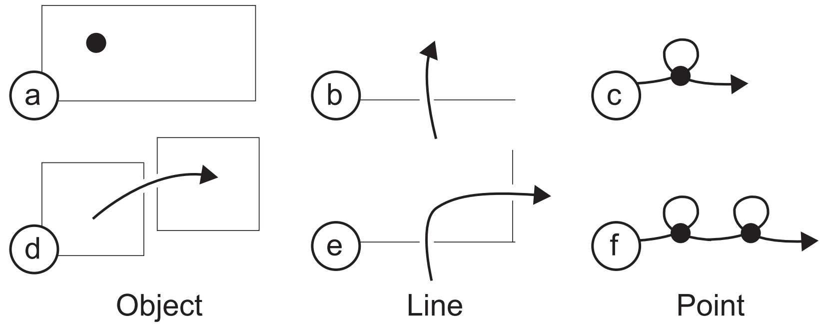

Figure 22: constructable allows users to select (a) objects by pointing, (b) lines by crossing, and (c) points with pigtails. (d-f) To allow for fast selection, all selection gestures can be extended across multiple objects, lines, and points.

implicitly, e.g., by the object being copied and does not require or allow for user input. However, to allow users to optimize material usage, we let users show constructable where to create the new object. As illustrated by Figure 23, users point constructable to available material by drawing a directional cropmark (see also *Papiercraft* [34]). The orientation of the cropmark specifies the orientation of the pasted object, allowing users to optimize for material use.

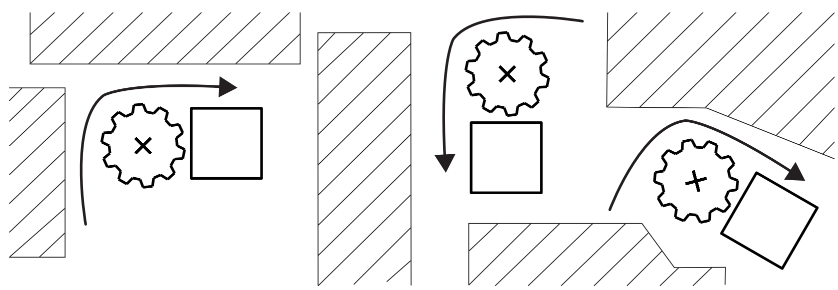

Figure 23: Users paste an object by drawing a cropmark. constructable places the object into the "inside" of the cropmark. Cropmarks allow users to place objects carefully into available space while preventing them from cutting into adjacent contents.

3.3 walkthrough: constructing a device

In the following, we illustrate constructable's tools at the example of the simple motorized vehicle that we had briefly touched upon in Figure 20. Figure 24 shows the final outcome and the pieces required to produce it. We start by creating the housing (Figure 25). (a) We use the *polyline* tool to sketch the rectangular base. (b) Using the sketch line button on the *scale* tool we create a sketch line rectangle around the base—this sets the height of the walls. (c) By crossing the north

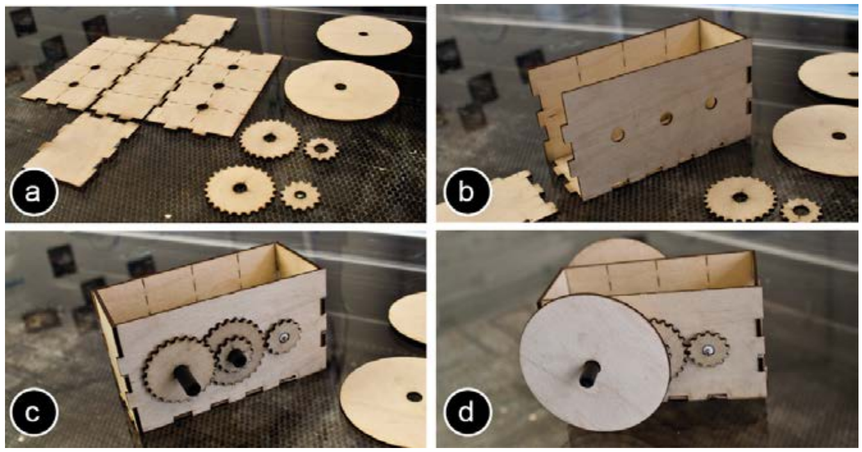

Figure 24: (a-d) The motorized vehicle in different states of assembly.

edge of the base with the *extrude* tool, we create the first wall segment. (d) For efficiency, we create the remaining three walls using a single long stroke that extrudes the base east, south, and west. There is no limit on concatenating, so we could have also extruded all four walls in a single stroke. (e and f) To allow us to assemble the housing later, we add finger joints. We connect the walls by crossing pairs of respective edges using the *finger joint* tool. (g) Finally, we assemble the box by connecting the finger joints.

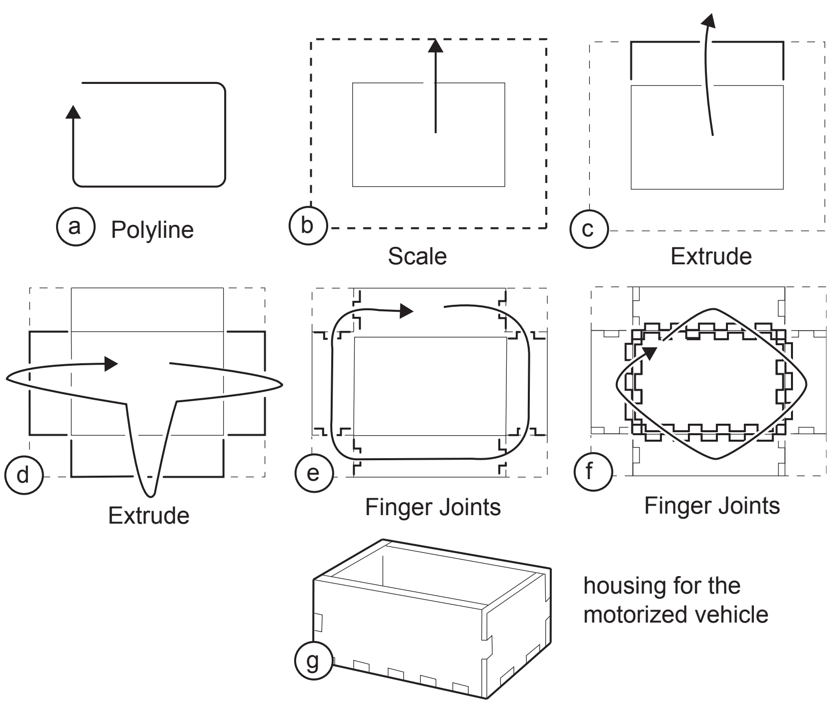

Figure 25: Interactively constructing the housing for the motorized vehicle.

Figure 26 shows how we add gearbox and wheels. (a) To make sure that we end up with straight axles we draw three sketch lines using the *polyline* tool. (b) We create the first axle hole using the *circle tool*, the location of which snaps to the intersection of the two sketch lines located close by. (c) We draw all remaining axle holes using a single stroke concatenating multiple pigtails.

(d) We create the first pair of gears by selecting two axle holes with the *gear* tool, by defining the transmission ratio through marking the point where we want the two gears to meet, and by showing constructable where to create the gears using a cropmark. We create the second set of gears accordingly. (f) To create a wheel, we first create an axle hole using the *circle* tool. We then create a wheel around this axle hole using the *scale* tool. (g) We create a second wheel by copying the first one using the *copy* tool.

We are now done creating our parts. We remove them from the machine and assemble them, resulting in the vehicle shown in Figure 24. The device is mechanically functional, because every mechanical connection was created using tools that embody the appropriate constraints.

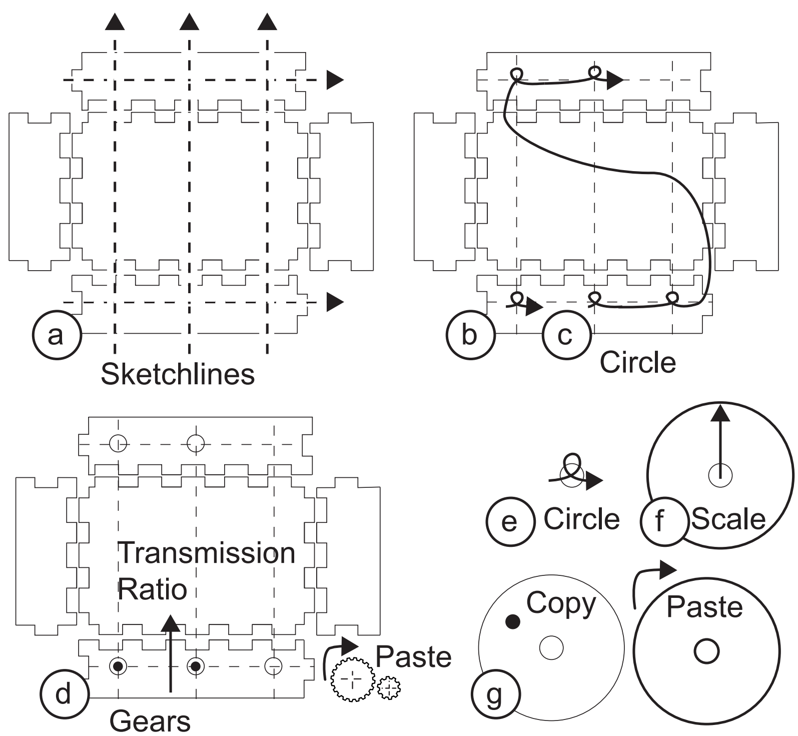

Figure 26: Continuing the previous example, we add axles, a two-stage gearbox, and wheels.

3.3.1 *Decorative functionality*

While constructable was designed with the goal of supporting interactive construction, its underlying concept of tools allows us to integrate form-giving (Figure 27) and decorative functionality (Figure 28) as well. Figure 27 shows how we create a wooden booklet that has round corners and a flexible bend in the middle.

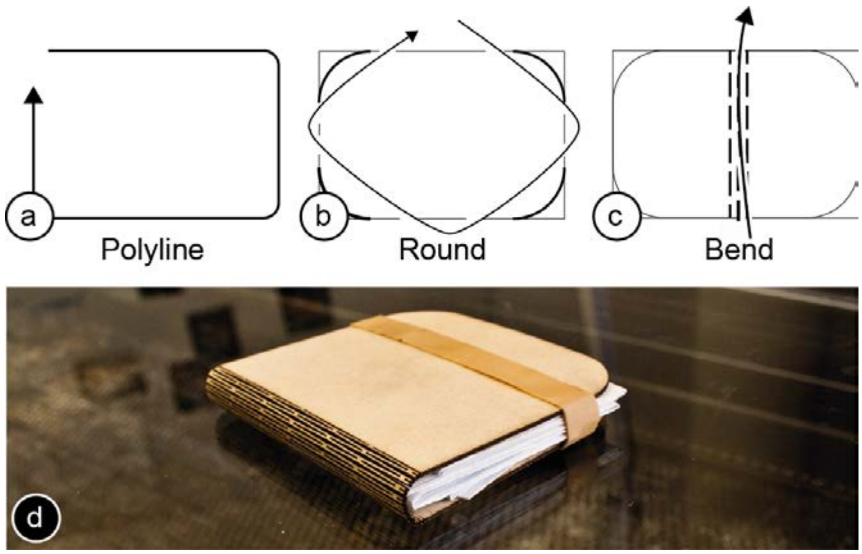

Figure 27: Creating a wooden booklet sleeve by (a) drawing the cover with the *polyline* tool, then (b) smoothing the corners with the *round* tool, and (c) making the wood flexible using the *bend* tool.

Figure 28 illustrates how we apply a picture to the housing of our motorized vehicle, á la CopyCAD [19]. (a) Continuing our earlier example, we place the housing of our motorized vehicle back into the machine, and position a photo with the desired logo on top of it. We wave the *rub-on* tool across those areas of the photo that we want

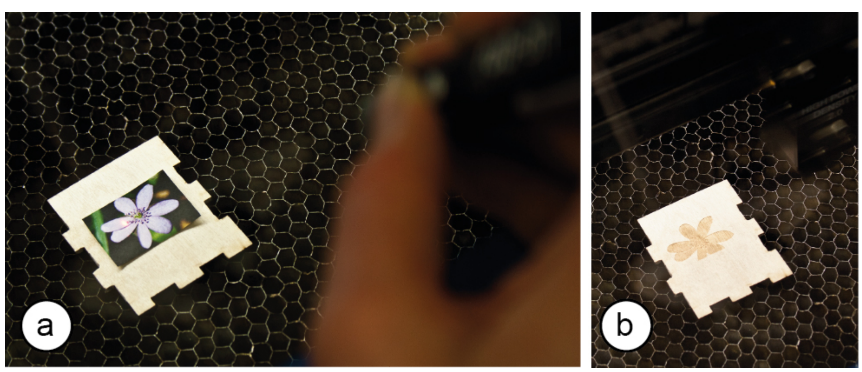

Figure 28: Adding a decorative logo onto the side of the motorized vehicle using the rub-on tool.

to transfer. (b) We take the photo back out, and as we close the lid, constructable engraves the logo into the housing.

Similarly, the *trace* tool cuts the contours of a physical object into the workpiece. Figure 29 shows how to use this to create a holder for two paint jars.

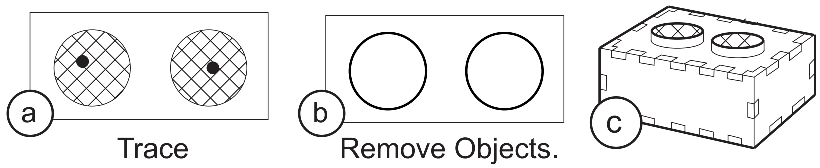

Figure 29: Creating a holder for two paint jars by (a) selecting physical objects inside the cutter using the trace tool, (b) After removing the physical objects constructable cuts. (c) Final jar holder.

Finally, we can use the *freehand* tool to create unconstrained freehand lines and cuts (Figure 30).

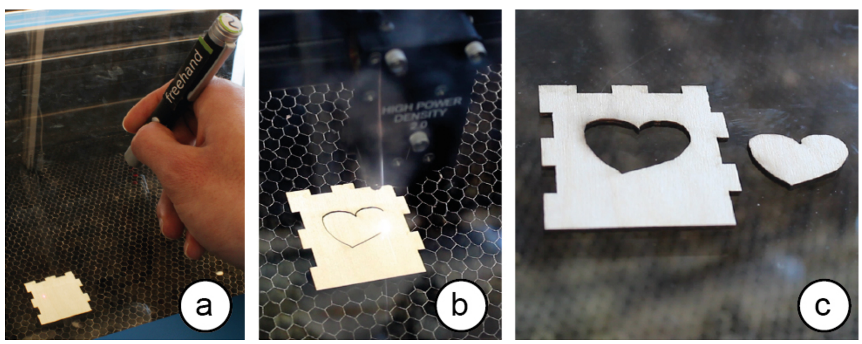

Figure 30: Sketching using the freehand tool.

3.3.2 *Trial-and-error support using "undo" tools*

Finally, constructable offers basic support for trial-and-error by providing an approximation of "undo" tools. Since physical cuts cannot be undone, constructable's "undo" tools instead refabricate the object — they create a copy that does not have the cut. Users apply the tool by crossing the cuts they want removed; they then paste the newly restored object. While the primary purpose of the tool is to repair and undo, the way it achieves this is by uniting two objects and copying the result. Since this functionality is useful beyond undo, we ended up giving the tool the name union tool.

In practice, any tool that unites two objects can be used as an undo tool. Figure 31 shows the *butterfly joint* tool, which connects two objects using a butterfly connector. This tool produces a butterfly-

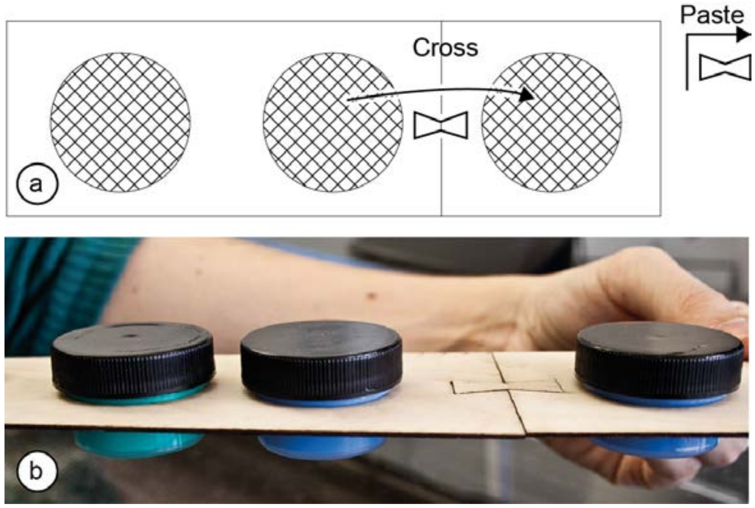

Figure 31: Undoing a cut by re-uniting the pieces using the *butterfly* tool.

shaped hole across the cut and lets users paste a matching butterfly-shaped connector. Users repair the cut on assembly by placing the connector into the hole—it sits tight enough to create a lasting connection. While the *union* tool obviously creates the stronger connection, a butterfly joint consumes less material.

3.4 BENEFITS

In the above walkthrough, we demonstrated interactive construction using constructable. We showed how constructable allows to interactively construct a functional mechanical device by bringing some of the key qualities of CAD-based personal fabrication into interactive fabrication.

3.4.1 *Precise input and output*

constructable achieves the precision required to make functional mechanical devices as follows:

Precise input: Even though all input to constructable is mediated through a hand-held tool, the resulting jerkiness never interferes with precision because all relevant parameters are appropriately constrained. Only parameters with no functional implications, such as the size of the base of the motorized vehicle, are defined freehand.

constructable uses three types of constraints: (1) Each tool implements constraints; the *polyline* tool, for example, draws only rectilinear contents. (2) Sketch lines allow users to add constraints explicitly. (3) A special class of tools that creates connections, such as finger joints create both halves of the connection at once. As a result, finger joints always fit perfectly even though they involve two parts. The *gear* tool is based on the same concept and one might say that it connects two axles.

Precise output: As mentioned earlier, all tools explain themselves exclusively through the cut or sketch line they produce. Since this allows all output to be created using the laser cutter itself, constructable's geometry output is extremely precise. This offers orders of magnitude higher resolution than projection and is never subject to calibration issues.

3.4.2 *No projection*

As can be seen in Figure 32, an early version of our system did feature a projector, similar to several systems in the related work, such as CopyCAD [19]. Letting go of it not only increased precision, but also made sure we directed users' attention at the workpiece at all times, rather than at a projection collocated with the workpiece.

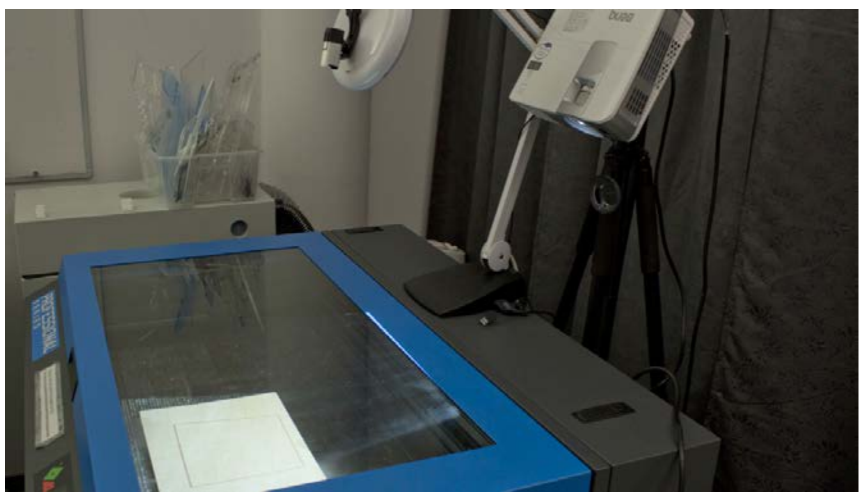

Figure 32: An earlier version of our system did feature a projector. However, we decided to solely use the laser cutter for all input and output.

A side effect of not having a digital display meant that we had to eliminate all hidden state and modes, because there was no way to keep users informed about them. This led us to the design of the proxy laser model, in which the current set of constraints is represented solely by which physical device the user is currently holding.

As another side effect of using the cutter as the sole visual feedback device, sketch lines became permanent. While users can erase sketch lines by replicating the final object using the unite tool, we think that users will typically choose to leave these lines in, the same way that designers leave sketch lines in to illustrate their process or even because they are aesthetically pleasing [11].

3.4.3 *Trial-and-error using "undo" tools*

We implemented trial-and-error using special "undo" functions, such as the union and butterfly joint tool that allow users to re-create the previous state of an object by re-fabricating it.

3.4.4 *Fast interaction via the proxy laser mechanism*

As mentioned earlier, constructable achieves fast interaction using the proxy laser mechanism, i.e., drawing with low-power laser tools, the effect of which is implemented by a high-power cutting laser. We found the resulting interaction to invite a powerful conceptual model, namely that the cutting laser amplifies the proxy laser, similar to how power brakes and power steering amplify the driver's muscle strength. We plan to develop this notion further in future versions of constructable, with the ultimate goal of causing at least the freehand proxy laser to invite the interpretation that it itself is cutting.

3.5 LIMITATIONS

Like any tool, constructable is useful for some design problems and less so for others. While constructable extends the range of problems that can be tackled interactively, it is obviously not a CAD system. As with traditional woodworking tools, some types of projects can be tackled tool-in-hand, while more advanced problems require users to sit down with a piece of paper first. The same way that saw and wood chisel cannot replace a detailed design process, constructable cannot replace CAD.

Another limitation is that all construction with constructable is inherently scale 1:1 and constructable offers no way of inspecting a detail in magnification. Similar to working with traditional woodworking tools, this limits users to projects that fit a particular scale.

Finally, constructable was not designed with walk-up use in mind. While some tools, such as the finger joint tool have the potential to make complex construction elements accessible to inexperienced users, mechanical construction in general does require know-how. Rather than addressing first time users, we designed the majority of constructable's tools so as to be generic in nature, to apply to each other, and thereby allow for a wide range of constructions.

3.6 ergonomics: the drafting table form factor

While constructable is primarily about constructive functionality, we made some observations on ergonomics.

While we initially perceived it mostly as a design hurdle, the laser cutter's glass cover turned out to become one of the key elements creating the affordance of our system. (Figure 33a). By allowing users to rest their body weight on the glass, it allows users to get even closer to the workpiece without worrying about interfering with it. Furthermore, we found ourselves resting proxy lasers on the glass while drawing (Figure 33b), which adds substantial stability, making the interaction even more precise.

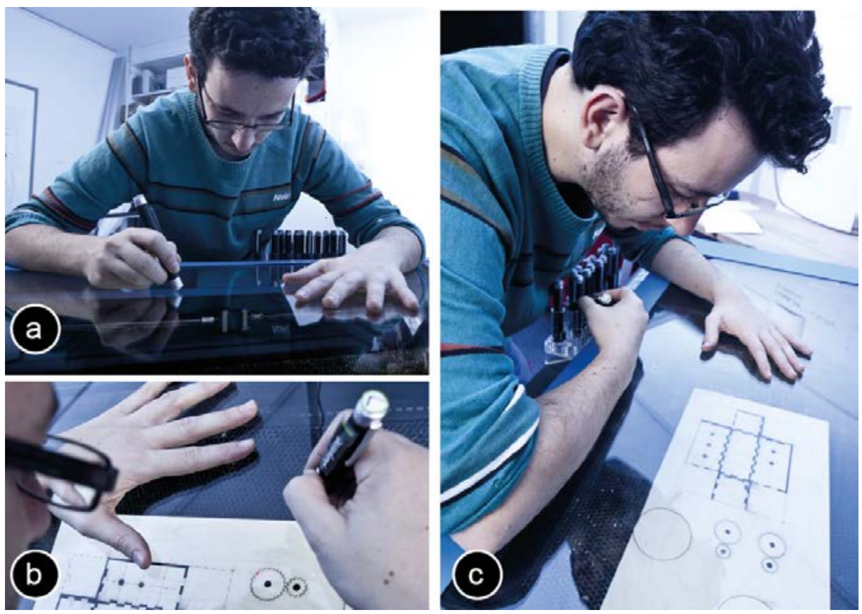

Figure 33: (a) The glass cover supports users' weight, allowing them to get close to the workpiece. (b) Resting proxy lasers onto the glass allows for precise interaction.

To invite this interpretation and posture, we positioned the proxy laser tools as shown in Figure 33c. This allows users to reach tools without lifting their arm, but instead pivoting around their elbows similar to the Lagoon in Alias Sketchbook [18].

We found all of the above to invite the interpretation of constructable as a drafting table, the "drawing" on which is the actual physical object itself. We plan on building on this interpretation in future versions by tilting the glass cover slightly towards the user (e.g., as in ActiveDesk [17]).

3.7 contribution

The main contribution of this paper is the concept of interactive construction, which we support with the functional system prototype constructable. Our demo scenarios illustrate how users can create functional mechanical devices, which exceeds the complexity of examples created in the interactive fabrication literature. We are thus primarily making a systems contribution.

In addition, our system contains a series of novel elements, including (a) the proxy laser mechanism, i.e., drawing with low-power laser tools, the effect of which is implemented by a high-power cutting laser, (b) the use of the laser cutter as a super high precision display, replacing projection, and (c) the notion of undoing a physically destructive action by re-fabricating the broken piece.

3.8 implementation

Figure 34 shows how constructable processes proxy laser input in order to generate cutting laser output.

constructable has two main components: the first covers all the data processing and is written in C++, the second covers all the laser-cutter instructions and is written in Java. constructable uses the following libraries to achieve its functionality: (1) OpenCV for computer vision, (2) PaleoSketch for shape recognition, (3) OpenOffice API for creating the laser-cutter document, (4) OSC for the communication between its components, and (5) QT for its graphical user interface, which is only used for debugging purposes and to adjust the tracking settings.

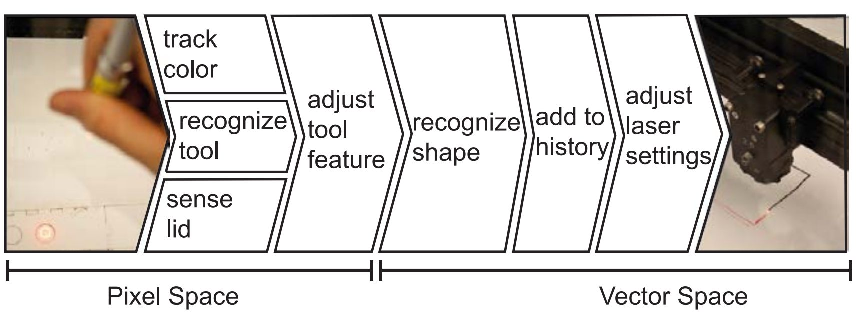

Figure 34: Data processing flow in constructable.

3.8.1 *Sensing*

constructable observes the workpiece using a web cam (MS Lifecam, 844x448px, 30 fps). Since the camera image needs to be aligned with the rectangular laser cutter, constructable calculates a homography. As homography input, we manually indicate the four corners of the laser-cutter in the captured image by clicking on them. Once the homography is calculated, we apply the resulting transformation matrix to each frame before further processing it.

constructable extracts the red dot produced by the laser-pointer using color tracking (Figure 35a). For this, constructable first converts the image into the HSV color space. It then provides the user with a user interface to filter for the bright red laser dot using sliders to set minimum and maximum values for the hue, saturation, and value channels. When the user is done, constructable writes the current settings into a log file, which is automatically loaded when constructable is started the next time.

constructable only starts tracking the red laser dot, when the cut or sketch line button on the laser pointer is pressed (Figure 35b). To extract the red laser dot from the image, we first threshold the HSV image using the minimum and maximum values for each channel, which results in a binary image. Afterwards, we run the connected component algorithm and identify the largest component as the laser dot. We now fit an ellipse around the laser dot and use the center of the ellipse as the tracking coordinate. While the user keeps the cut or sketch line button pressed, we repeat the process on every frame and write the coordinates into a list.

When the user releases the cut or sketch line button, constructable smoothes the path using the Kalman Filter. It then performs shape recognition on the path using the PaleoSketch shape recognizer [43] (Figure 35c). As the sketch recognizer is only available in Java, constructable sends the path data via the Open Sound Protocol to PaleoSketch and PaleoSketch returns the shape after processing is done. PaleoSketch does not return a single shape, but a list of shapes with the best fit at the top of the list. Since all constructable input is based on lines (i.e. a pigtail is also a set of short lines), we only activate the LineTest and the PolylineTest in PaleoSketch and deactivate all other possible shapes, which increases recognition accuracy.

After the drawn path is converted into a geometrical primitive, constructable applies tool specific geometric operations (Figure 35d). For instance, when the user crosses an edge with the *round corner* tool, we calculate the intersections of the drawn line with the existing geometry. If two connected lines are crossed, the user crossed a corner. We then calculate the corresponding bezier-curve points at the correct lo-

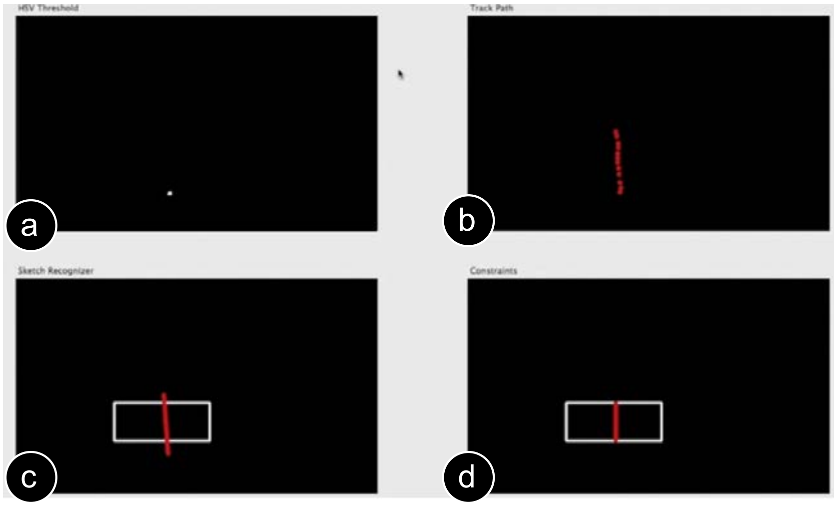

Figure 35: constructable processes input by: (a) thresholding the camera image to extract the laser dot, (b) tracking the laser dot over time, (c) performing shape recognition on the list of points, and (d) applying tool specific operations, such as calculating intersections.

cations. Another example is the *extrude* tool. If the user extrudes an edge, we take the endpoint of the path, calculate the distance from the endpoint to the edge, and then use this information to construct the new line parallel to the existing line.

At this point, constructable records the interaction history, which it uses to support undo and selective repairs using the union tool.

3.8.2 *Output to the laser cutter*

constructable now outputs the shape to the laser cutter, currently a Universal PLS6.150D. Using the OpenDraw API, constructable draws the shape into an OpenDraw document that has exactly the size of the cutting table (Figure 36). For this, constructable first initializes a connection to the OpenDraw program, it then loads the OpenDraw document, and then calls the different shape factories for creating and drawing the shapes. According to the available tools, constructable currently supports circles (for the circle tool), beziers (for the round corner tool), point lists for any polygonal shape (for, e.g., the rectangle or trace contour tool) and images (for the rub-on-texture tool). Before the shapes are placed into the OpenDraw document, constructable transforms their position from the camera space (844x448px) to the document space (80x45cm) using a constant scale factor.

After constructable placed all shapes into the OpenDraw document, it sends the document to the laser cutter using the regular printer interface. As required by Universal Laser Systems, constructable encodes the necessary meta-information into the color of the respective

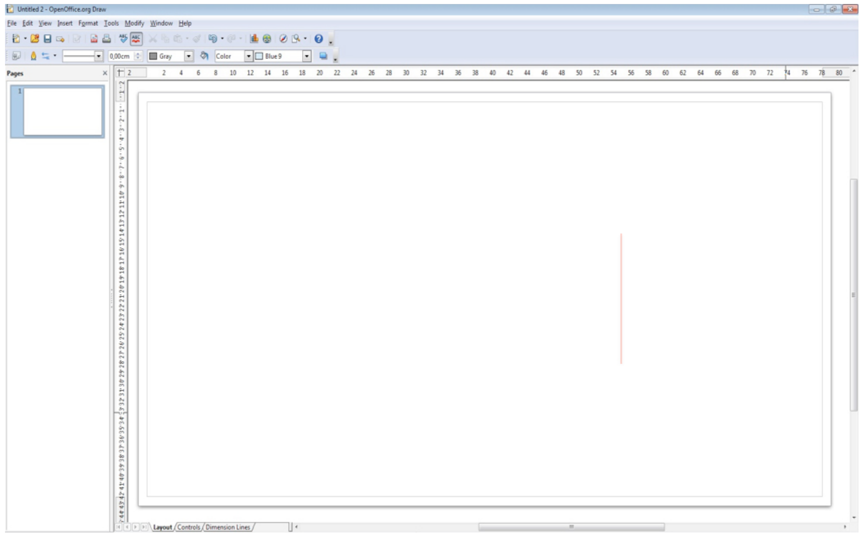

Figure 36: constructable automatically draws the geometry in an OpenDraw document and sends it to the laser cutter.

line, i.e., cutting-depth (mm), laser power (percentage), and speed (percentage). A red line, for example, causes the laser to cut, while green is used to create a shallow, low-power sketch line. constructable sends all communication using the Open Sound Protocol, which makes it easy to adapt the system to other hardware components, such as a different laser cutter model.

When the document was successfully sent to the laser cutter, we use a mouse robot to automatically press the start button in the laser-cutter software (Figure 37). While this is the only working approach to control the laser-cutting process via code, it also has the drawback that the laser-cutter user interface always need to be the active window and that the window needs to be full-screen as the mouse click

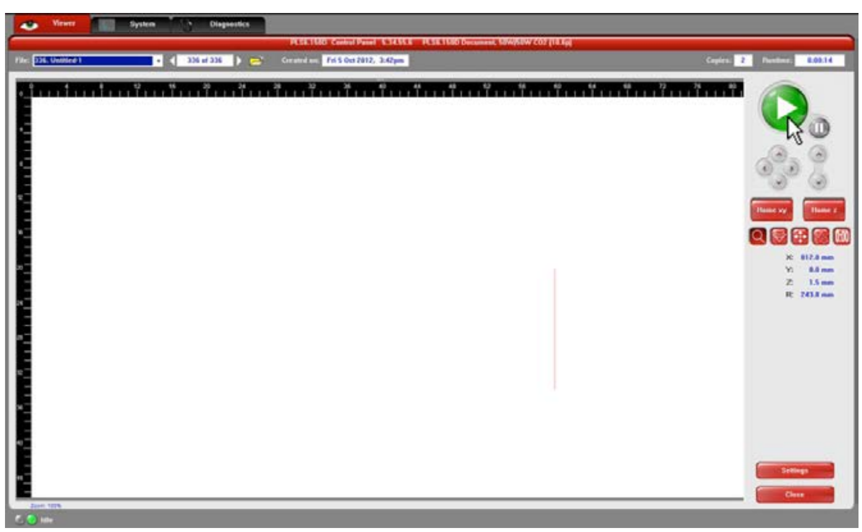

Figure 37: constructable uses a mouse robot to start the cutting process.

position is in absolute coordinates. We are currently in contact with Universal Laser Systems, which will provide us with an API to control the laser cutter in the future.

3.8.3 *Proxy lasers*

In order to retrieve the constraint set represented by the current tool, constructable determines which tool is in use using mechanical switches, one of which is installed in each tool holder (similar to pen "recognition" in SmartBoards, smarttech.com). This bypass allows us to implement all proxy lasers using a single type of off-the-shelf laser pointer. The switches in the toolbox are connected to an Arduino Uno, which runs a program to determine which switch was activated and if the tool was taken out or put back in (Figure 38). The current state of the toolbox is then communicated to the main constructable application via the Open Sound Protocol.

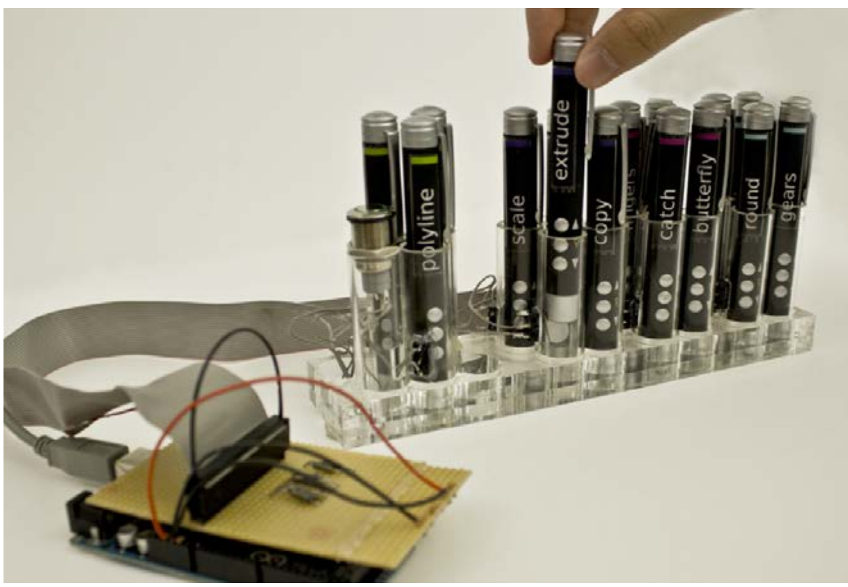

Figure 38: We determine the active proxy laser using mechanical switches in each tool holder.

The buttons on all proxy lasers trigger an infrared signal, which constructable monitors using an IR receiver placed next to the camera. By default, the commands sent by the laser pointers are mapped to the left and right arrow key, as they are normally used in presentations to forward to the next slide or to go on slide back. constructable, in contrast, maps these key commands to the cut and sketch line mode using QT's key API.

The cut and sketch line buttons are spring-loaded in order to eliminate mode errors. Users operate them without letting go of the tracking button. For optimized ergonomics during prolonged use, constructable offers a dual-footswitch pedal, which is fulfilling the same purpose as the cut and sketch line barrel buttons. When the footswitch is pressed, a program written in PureData notices the change and sends the current state to the main constructable application via OSC.

3.8.4 *Remixing: using physical objects as a reference*

constructable captures physical objects using the same camera that tracks the laser dot. A bar of fluorescent light mounted inside the cutter supports this by providing homogeneous, reflection-free illumination of the workpiece. We mounted the bar of light inside the laser cutter by attaching magnets to its backside. We also guided the cable in a way that it would not interfere with the laser beam that exits the laser module at the top left corner of the laser cutter.

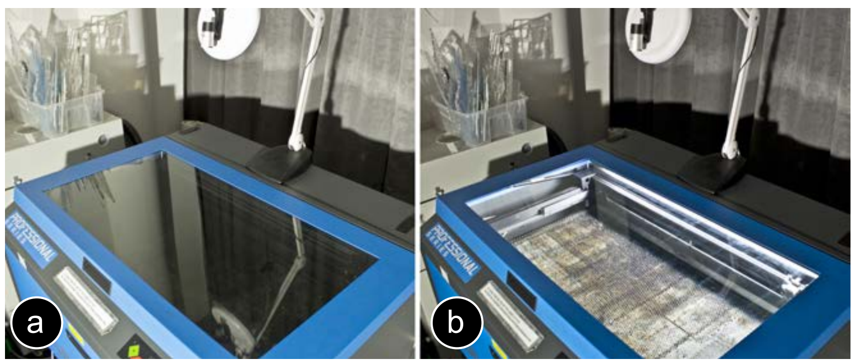

Figure 39: (a) To minimize reflections on the glass surface, we (b) mounted a bar of fluorescent light into the laser cutter.

To detect if the user inserted a physical object for remixing, there is no need to continuously track the interior of the laser cutter as objects can only be inserted or removed if the lid of the laser cutter was opened. The laser cutter already provides a mechanism for detecting the lid state for safety reasons, i.e. it stops cutting as soon as the lid is not completely closed. However, without a library to access this built-in feature we were not able to integrate it into constructable. Instead, we placed an IR rangefinder sensor behind the lid that informs the system when the lid is open. When the lid opens, the distance to the range finder gets smaller – when the lid closes again it gets larger. We connected the range finder to the same Arduino Uno that is used by the toolbox. As all other components, it communicates its state to the main constructable application using the Open Sound Protocol.

In the following section, we describe how we use OpenCV for copying object contours. After the user inserted physical objects, closed

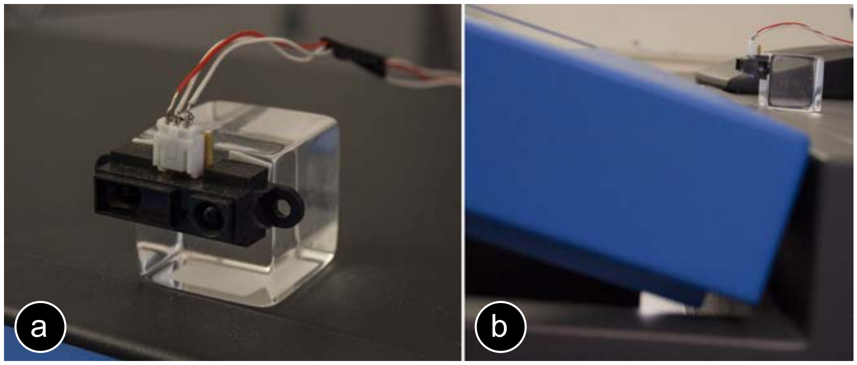

Figure 40: (a) We use this IR range finder to (b) detect if the lid of the laser cutter lid is open or closed.

the lid, and used the trace tool to indicate that the user wants to copy the objects, we start the processing pipeline. First, we do background subtraction: we use the current frame as the input and a frame when no object was in the laser-cutter as the background. Now we extract all contours using the connected components algorithm and remove contours that have an area smaller than a certain threshold as noise. When the user opens the lid and removes some objects, we repeat the process, but at the end check if any contour disappeared. The disappeared contours are the removed objects and are laser cut as soon as the user closes the lid again. The remaining contours are on hold as we do not want the laser cutter to cut in places where objects are still present.

3.9 evaluation: demo at tei'13 and chi'13

We gave a live demo of constructable in February at TEI'13 [56] in Barcelona and in May at CHI'13 [13] in Paris. The demo session at TEI'13 was one afternoon from 3pm to 8pm, the demo session at CHI'13 was two full days parallel to the normal paper track. During these two demos, we had around 60 conference attendees directly interacting with constructable by fabricating their own object and several hundred bystanders watching and commenting on the system.

Before demoing constructable, two students (Konstantin Kaefer, Bastian Kruck) helped to convert the research prototype of constructable into a system that could be used by any conference attendee. Demoing constructable also requires at least three people: one observing the constructable system on the laptop, one guiding the currently interacting attendee, and one answering questions of the bystanders.

Figure 41: We transported the laser cutter to the demo venues in a van.

3.9.1 *Preparation and demo setup*

We drove our own equipment including the laser cutter and its filter to the demo venues in a small van. Before we lifted the laser cutter into the van, we removed all sensitive parts, such as the two laser modules, from the cutter and packed them into soft blankets. This also reduced the weight of the laser-cutter from 156kg to 124kg since each laser module on its own weights 16kg. We also removed the honeycomb from the laser cutter and attached all moving parts in the laser cutter with tape.

The demo setup consisted of the different constructable components (laser cutter, overhead camera, toolbox, the infrared receiver, and laptop), as well as additional information material, such as a paper board explaining the tool-buttons and paper boards with walk-throughs for constructing different objects.

Figure 42: (a) Demo setup at the FabLab Barcelona. (b) Final setup.

Attendees were invited to either construct one of the two example objects by following a walkthrough (the wooden booklet or a small storage box) or to create an object on their own.

Figure 43: (a,b) More than 60 conference attendees used constructable to create their own object. (c) Several hundred bystanders watched them interacting.

3.9.2 *User feedback*

While we did not conduct a formal user study, we gained many insights into how constructable can be improved by watching the conference attendees interact, by speaking to them, and by overhearing the conversations of bystanders. While every attendee who used constructable was able to finish one of the objects under our guidance, there were also the following areas for improvement:

(1) Using a springboard instead of three barrel buttons: Our current interaction model requires the user to press two barrel buttons on the laser pointer at the same time - the middle button for turning on the laser-pointer and the front or back button to start the recording of the drawing. Several attendees released one of the buttons accidentally during drawing because their thumbs were not able to hold the two buttons pressed for a longer period of time. In the future, we want to solve this problem by using a springboard [24] on top of the three barrel buttons (Figure 44).

(2) Using cameras inside the laser cutter to prevent occlusion: constructable's camera is mounted on top of the laser cutter. While the camera is slightly tilted to minimize occlusion with the user's hand, this approach cannot completely eliminate occlusion. Depending on how attendees were holding the laser pointer and how much they leaned towards the laser cutter, we experienced that sometimes constructable did not capture any data because the laser dot was not

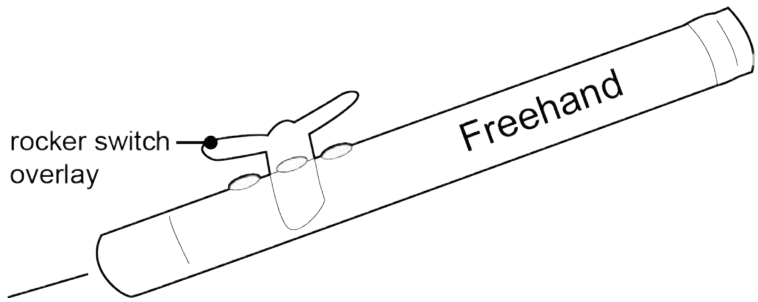

Figure 44: This rocker-switch overlay improves ergonomics during prolonged use.

visible to the camera. For future work, we want to examine if it is possible to position the camera directly inside the laser cutter, which would eliminate all occlusion problems.

(3) Parameter entry: Several attendees expressed the need for entering parameters into the tools. For instance, when creating the rounded corners of the booklet, several attendees stated they would like to adjust the corner radius, which currently has a fixed size.

(4) Providing real-time feedback: Some attendees stated that they would like to have real-time feedback, for instance, to see the draw path progressing while they are drawing a rectangle with the laser pointer – and not just a single laser dot. As constructable provides feedback through laser cut sketch lines, real-time feedback requires to have the laser cutter act in real-time. Unfortunately, our Universal Laser Systems laser cutter can only be operated through the printer interface and is thus to slow to follow the proxy laser immediately.

Figure 45: The LaOS board allows us to send commands to a laser cutter in real-time and thus allows us to explore real-time visual feedback.

However, we found an open source control board called the LaOS board [54] that actually allows for real-time commands based on the machines low-level G-Code instructions. The LaOS board can be easily integrated into the LS3020 laser cutter from HPC LASER LTD and thus provides a good platform for exploring real-time feedback in interactive laser cutting.

3.10 CONCLUSION

constructable is an interactive construction system that enables users to interactively create functional mechanical devices. However, since constructable is based on a laser cutter it can only create two-dimensional parts, which later on need to be assembled into 3D objects.

For the interactive construction of three-dimensional objects, a 3D printer seems to be the tool of choice as it offers the most freedom of shapes. The problem with current 3D printers is that they are too slow for interactive construction as they build up the object one voxel at a time.

Therefore, we developed a novel laser technique called LaserOrigami for creating three-dimensional objects with a fast laser cutter. LaserOrigami achieves three-dimensionality through cutting and bending the workpiece in a single integrated process. To allow for interactive construction, we embedded LaserOrigami into constructable.

4 LASERORIGAMI: LASERCUTTING 3D OBJECTS

When fabricating three-dimensional objects, additive fabrication methods, such as 3D printers, offer the most freedom in the shapes they can produce. However, as they assemble objects from individual voxels, the time required grows cubed in the size of the workpiece — thus they are slow. Laser cutters achieve much higher speeds by assembling the object from 2D plates rather than individual voxels. However, laser cutters generally achieve three-dimensionality through the use of joints. These require assembly, which introduces repetitive manual labor into fabrication. This also limits how fast designers can iterate.

With LaserOrigami, we demonstrate how to overcome this limitation. Our rapid prototyping system LaserOrigami allows users to laser cut 3D structures by bending the workpiece rather than by means of joints, thereby eliminating the need for manual assembly (Figure 46). Inspired by a technique used to shape steel [21], LaserOrigami allows users to create 3D objects in a single fast and integrated process. As we demonstrate in section "Interactive LaserOrigami", our approach is fast enough to even allow for interactive construction.

Figure 47 shows an example object created using LaserOrigami — a mobile phone screen cam. This example was fabricated using five "bends", one of the three basic elements of LaserOrigami, which we discuss in detail in section "The Design Elements of LaserOrigami".

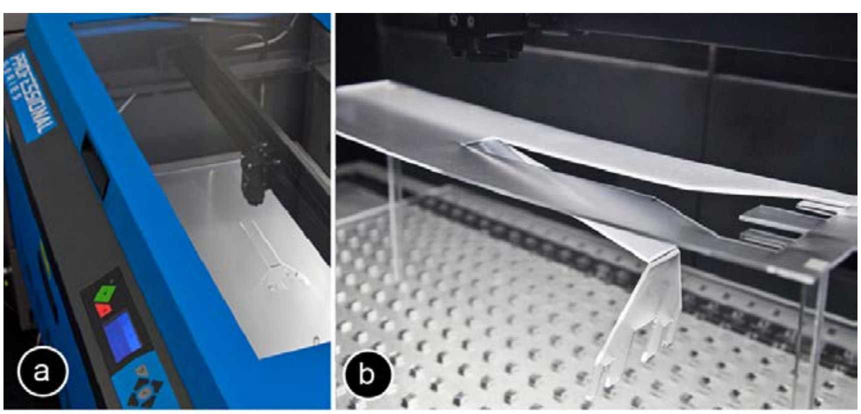

Figure 46: LaserOrigami fabricates 3D structure by bending, rather than using joints, thereby eliminating the need for manual assembly. It (a) cuts the contours and then (b) heats up the bend paths until the material becomes compliant and bends down under the influence of gravity.

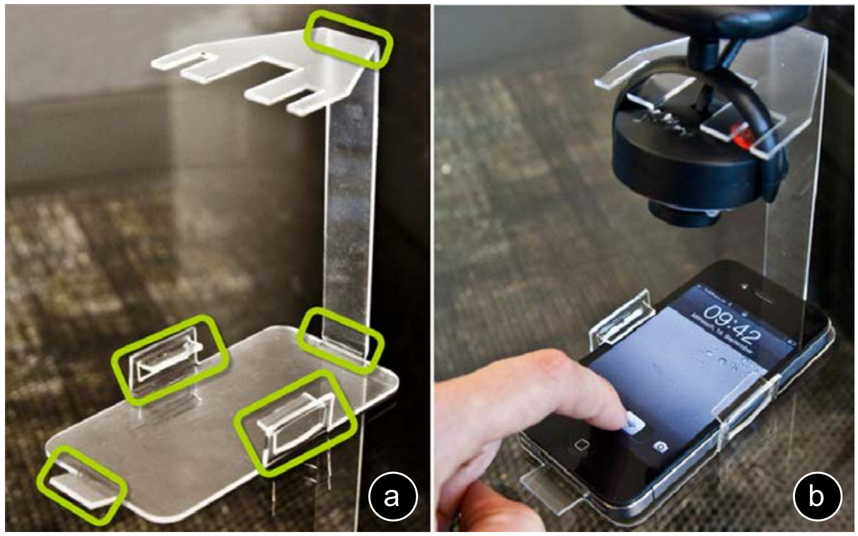

Figure 47: This mobile phone screen cam was fabricated using LaserOrigami. When the user retrieves the object from the laser cutter, it is already assembled and ready to be deployed.

4.1 BASIS OF LASERORIGAMI: BENDING WITH A DEFOCUSED LASER

The key idea behind LaserOrigami is that it achieves three-dimensionality by bending the workpiece, rather than by placing joints, thereby eliminating the need for manual assembly. LaserOrigami achieves this by heating up selected regions of the workpiece until they turn compliant and bend down under the forces of gravity.

LaserOrigami bends a workpiece by distributing the laser's power across a larger surface. As shown in Figure 48a, the cutting laser is normally focused on the workpiece, which causes the material to turn so hot that it evaporates.

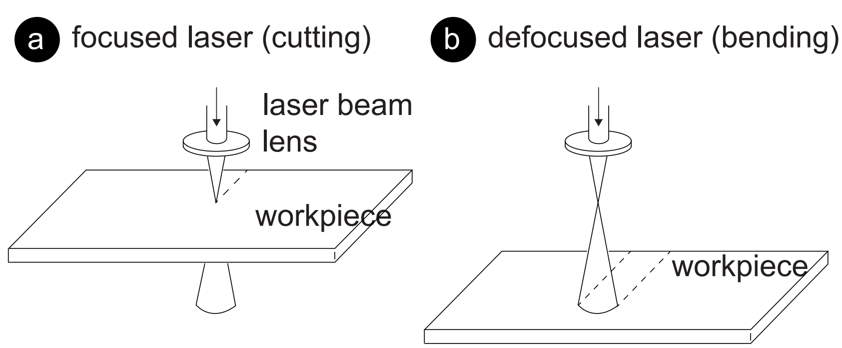

Figure 48: (a) LaserOrigami cuts the workpiece by focusing the laser on the workpiece, (b) it bends by defocusing the laser.

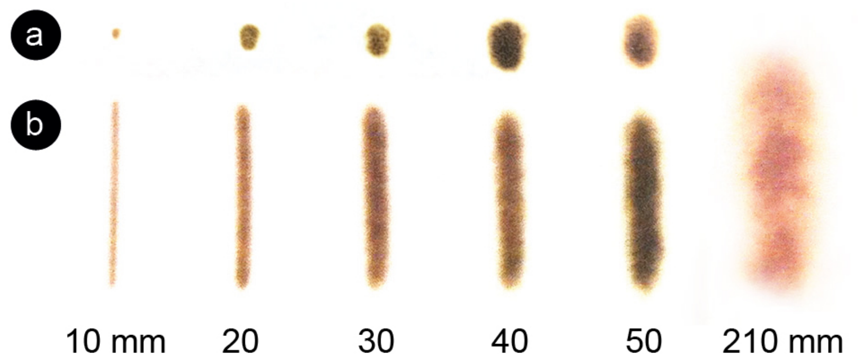

Figure 49: Heated area of the defocused laser when the motion table is lowered. (a) Laser pulses but does not move, (b) laser moves along a line.

In contrast, LaserOrigami bends the workpiece by distributing the heat over a larger surface. By moving the workpiece away from the laser it defocuses the laser (Figure 48b); this distributes the laser's heat over a larger region (Figure 49). In addition, LaserOrigami distributes the laser's heat further by repeatedly running the laser back and forth over the region to be bent. As a result, the workpiece heats up only to the point where it turns compliant; it then bends under the influence of gravity. The result is a precise 90° bend.

LaserOrigami modifies focus by moving the cutting table up and down; the cutter we used (model PLS6.150D) allows doing this under computer control. This allows it to implement cutting and bending in a single integrated process. When users take the workpiece out of the cutter, it is already fully assembled.

In order to allow parts of the workpiece to drop, we prop up the workpiece inside of the cutter (on our PLS6.150D laser cutter, this is possible after removing the honeycomb grid). We created a simple configurable support grid for this purpose (Figure 50).

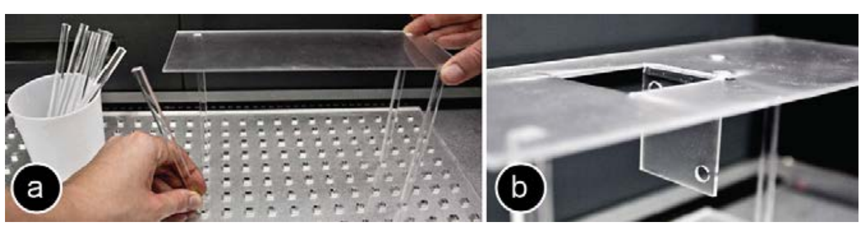

Figure 50: (a) This support grid (b) creates empty space below the workpiece that parts can fold and drop into.

4.2 laserorigami's cad user interface

LaserOrigami offers two types of interfaces. The first one is a traditional CAD-style interface, which we created as a master shape library for Microsoft Visio. Figure 51 shows a user creating the mobile phone screen cam from Figure 47. The user draws the object in Microsoft Visio with the LaserOrigami master shape library. To add a double bend, the user drags a double bend master shape into the design and rotates it to fit the existing drawing. The user repeats the process on the other side of the screen cam.

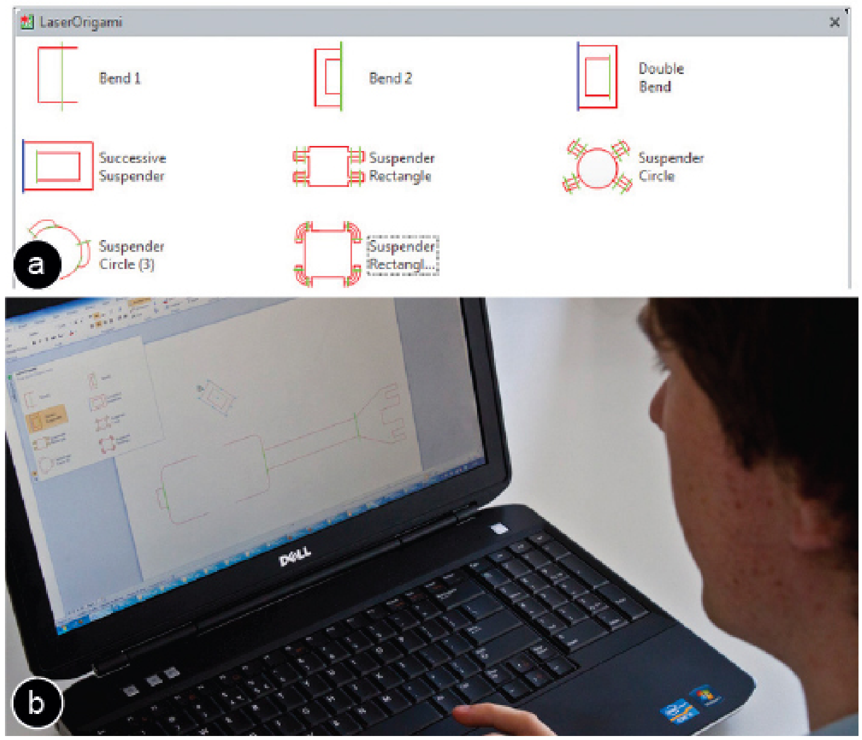

Figure 51: (a) LaserOrigami provides a Microsoft Visio master shape library for fast prototyping of folded objects. (b) Here the user has dragged a double bend from the LaserOrigami shape library onto the drawing surface and adjusts it by rotating the provided handle.

The shapes in the LaserOrigami master shape library encode all the "instructions" that the laser cutter requires in order to fabricate the respective shape, i.e., the lines that cut and the lines that implement the back-and-forth motion of the defocused laser. As shown in Figure 52b, we encode the back-and-forth motion of the defocused laser as pairs of lines of opposite orientation.

Switching between cutting and bending, i.e., moving the table up and down, is encoded in the line colors. As an example, Figure 52 shows the lines that implement a simple bend. In the configuration di-

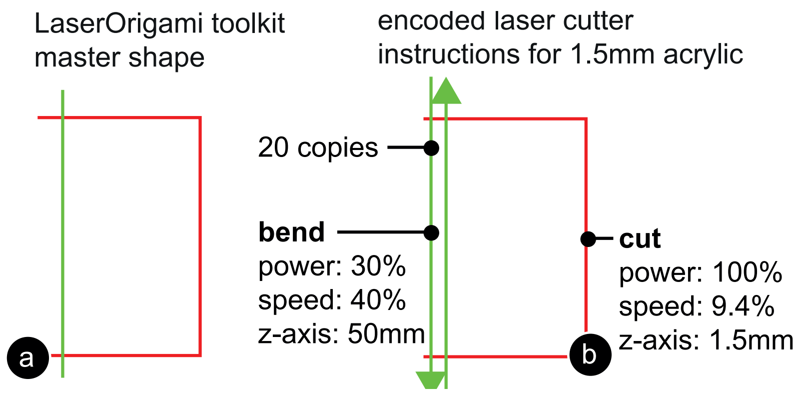

Figure 52: (a) This master shape from the LaserOrigami Visio toolkit encodes a 10cm bend for 1.5mm acrylic.

alog of our cutter, we configured red lines to mean cutting, i.e., whenever the laser encounters a red line the table will move the workpiece into focus. In contrast, we configured green lines so as to move the table down, causing the laser to go into defocused mode and heating up the material for bending. The property we manipulate here is called z-axis for our PLS6.150D laser cutter. It is normally used to move materials of different thicknesses into focus; with LaserOrigami we instead use it to defocus.

To make sure all features are executed in proper order, we arrange the stacking order of all lines within Visio; bottom lines are executed first, so we can arrange the cutting order using "send forwards/backwards" commands when creating the master shapes. Disabling the Vector Optimizer feature in our PLS6.150D cutter makes sure that line order is maintained during cutting. Also, since this cutter model always executes all lines of one color before moving on to the next color, we use a new line color for each group of cuts or bends.

Finally, we hide all lines of a master shape that encode laser cutter instructions by moving them on a hidden layer in the drawing. This allows the user to work with only the information necessary, i.e., to determine where cuts and bends will be executed by the laser but not how, thereby preventing a cluttered interface.

4.3 CONTRIBUTION, BENEFITS, AND LIMITATIONS

The main contribution of LaserOrigami is the concept of rapid prototyping of 3D objects using a laser cutter, so as not to require manual assembly. As illustrated by Figure 53, our approach is substantially faster than traditional 3D fabrication techniques such as 3D printing and unlike traditional laser cutting, the resulting 3D objects require no manual assembly.

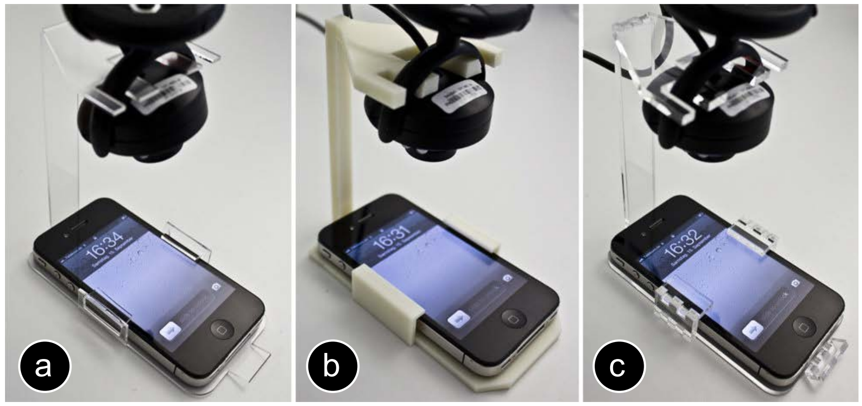

Figure 53: (a) LaserOrigami fabricated the screen cam from Figure 47 in 3min compared to (b) the 240min the 3D printer required (Dimension Elite 3D printer). (c) Traditional laser cutting requires assembly.

We introduce three design elements (bend, suspender, and stretch), we extend the design space by adding a servomotor, and we offer two user interfaces, i.e., a traditional CAD-like interface through Microsoft Visio and an interactive fabrication interface through constructable.

The bending concept behind LaserOrigami offers the following four advantages. (1) Faster than 3D printing, eliminates need for assembly from laser cutting. (2) Sturdier result than finger joints, because bending allows fabricating objects in one piece. (3) Easier calibration: traditional finger joints require a very precise calibration to achieve the desired tight fit; bending does not. (4) Cheaper than 3D printing.

On the other hand, LaserOrigami is also subject to four limitations. (1) LaserOrigami is limited to object shapes that can be constructed by cutting, folding and stretching the material. (2) Works only with materials that become compliant when heated up. (3) Limited length of what can be bent or suspended in one piece. If a bend is too long, it cools down faster than the laser can heat it up. (4) Limited material thickness, again limited by the power of the laser.

4.4 RELATED WORK: CUTTING AND FOLDING 3D OBJECTS

Origami techniques [26] demonstrate how to build thousands of different objects by combining various paper folding techniques. Adding electronics to these paper-crafted objects allows to quickly prototype interactive devices [49], to actuate single objects [45], and to even animate entire pop-up books [46].

To deform more rigid materials, researchers have developed a range of approaches based on compound materials and hybrid fabrication.

To make 3D prints foldable, Deng et al. [15] add silicone to the 3d printed structures. To achieve a similar effect with lasercutting, Balkcom et al. [6] sandwich foil between stiff poster boards. Liu et al. [35] use 2D polymer sheets with regions of black ink, which shrink and relax when absorbing light, thereby working as hinges. Pop-up MEMS [57] finally shows how to achieve complex 3D microstructures by bonding multiple layers.

Other approaches have been developed to handle materials that become compliant when heated up, such as acrylic. To shape acrylic, traditional shops contain devices like heat guns (emit a stream of hot air) and strip heaters (consist of a line of heating elements). Both have to be operated manually by the user, which can be cumbersome during design iteration. Researchers in material science have shown how to deform metal using a laser [21] by causing it to locally expand and contract, which the sheet metal evades by bending. They showed how to build simple concave [7] and convex shapes [33], and how to reshape existing 3d objects [32].

LaserOrigami builds on this concept in that it heats up materials using a laser. It exploits a different effect though, in that it heats up material to achieve compliance, not internal tension; this also allows LaserOrigami to work with acrylic and is an order of magnitude faster.

4.5 THE DESIGN ELEMENTS OF LASERORIGAMI

LaserOrigami assembles all designs from the three basic design elements shown in Figure 54, i.e., (1) the bend, (2) the suspender, and (3) the stretch.

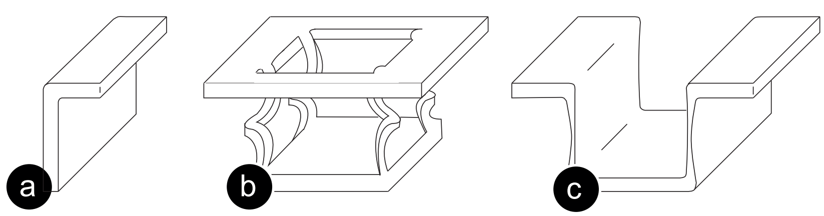

Figure 54: The design elements of LaserOrigami: (a) the bend, (b) the suspender, and (c) the stretch.

4.5.1 *Bends*

We already demonstrated bends in Figure 1. Bends also allow fabricating decorative elements, such as those shown in Figure 55.

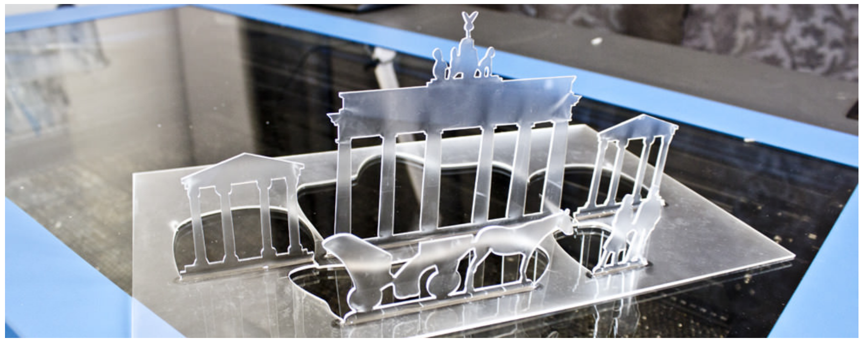

Figure 55: Using bending to create this decorative city outline.

The bend element shown above only allows bending up to 90°, which limits our designs to 2.5D. We can bend past the vertical axis by holding the workpiece slanted against the desired bend direction (Figure 56).

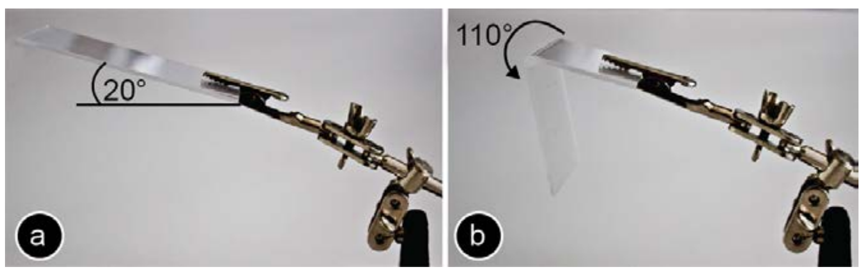

Figure 56: By slanting this workpiece 20° against the direction of gravitational force, we achieve a bending angle of 110°.

A more flexible solution is to bend successively (Figure 57). The purpose of the outer patch is to serve as a lever—once it has done its job we will typically cut it off. The lever approach thus obviously comes at the expense of material.

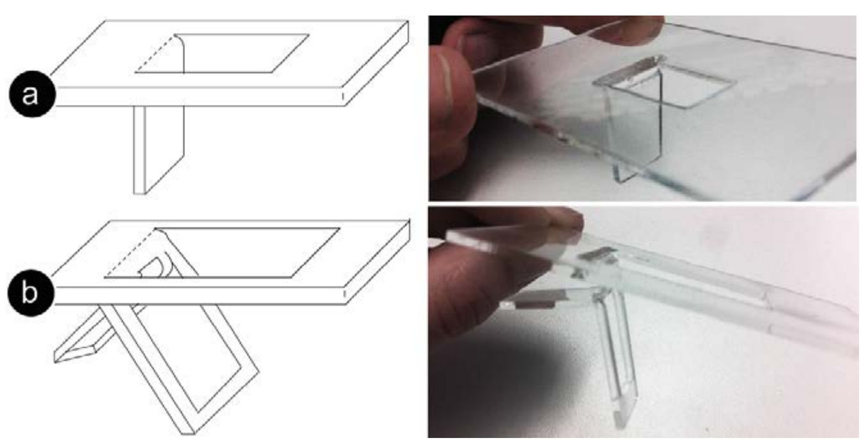

Figure 57: Successive bending allows the inner patch to bend beyond 90°.

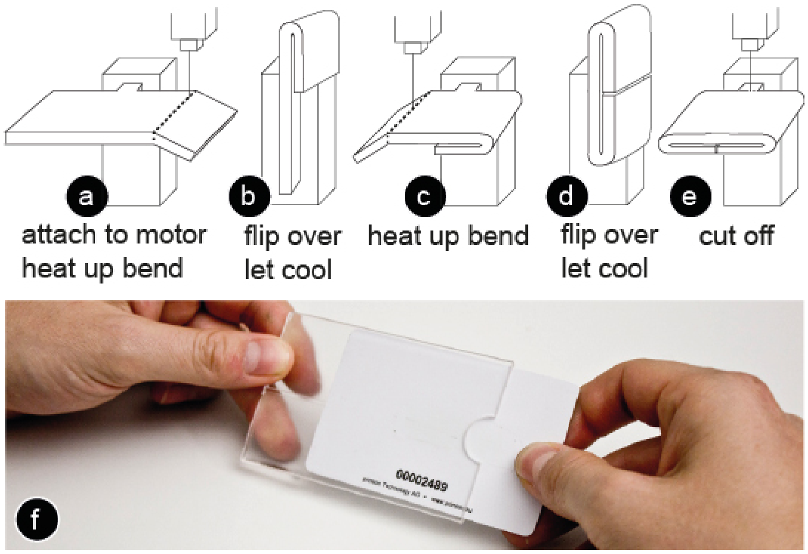

Figure 58: Fabricating this credit card holder using a servomotor.

A servomotor, finally, provides even more flexibility as it allows rotating the workpiece repeatedly (Figure 58).

A small tab that is part of the workpiece locks into the servomotor (Figure 59). When done, LaserOrigami cuts off the tab, causing the assembled workpiece to drop.

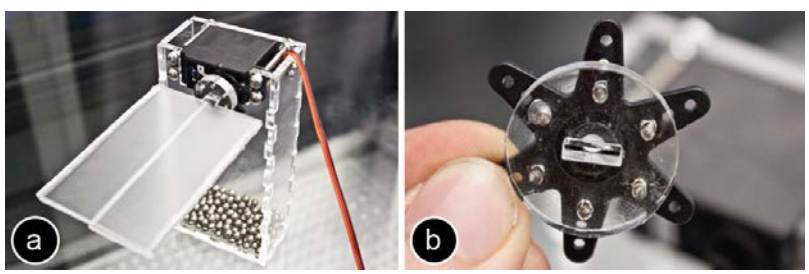

Figure 59: We attach the workpiece to the servomotor using this tab.

4.5.2 *Suspenders*

Suspenders allow suspending material in a controlled way. They are designed to unfold when heated up with the laser (Figure 60).

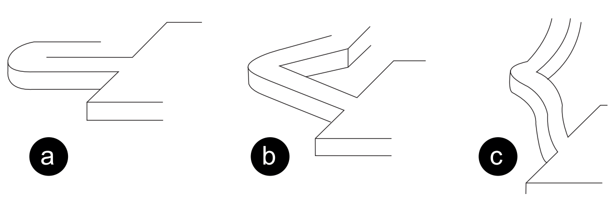

Figure 60: Suspenders: (a) the raw path that is cut.

The length of the suspenders defines how deeply the patch will be suspended (Figure 61).

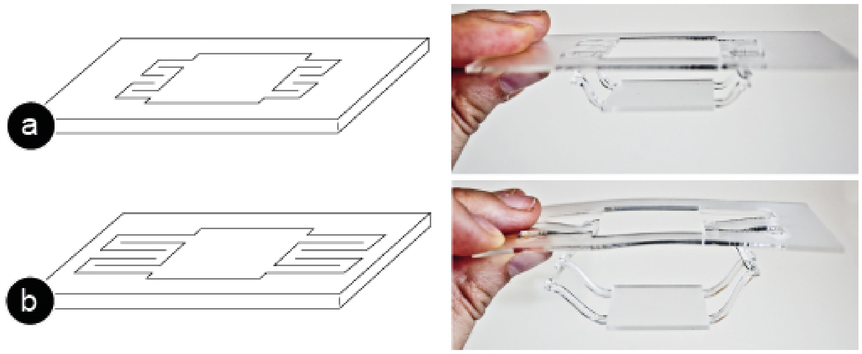

Figure 61: The same patch with (a) short and (b) long suspenders.

The use of three or more suspenders of identical length creates a horizontal patch (Figure 62a). While we will most commonly suspend objects parallel using suspenders of equal lengths, suspenders of unequal lengths allow us to create ramps (Figure 62b).

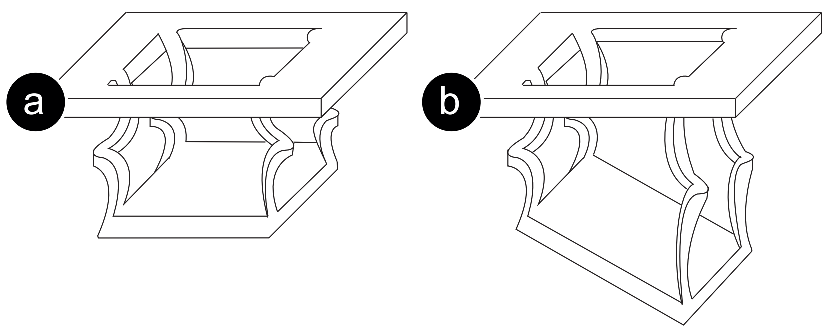

Figure 62: (a) Suspenders of equal length suspend a horizontal patch, (b) unequal suspenders create a ramp.

Suspending recursively allows us to create 2.5D landscapes from elevation lines (Figure 63a). To keep the next suspension in the plane of the laser cutter, we suspend the inner ring first, working outwards.

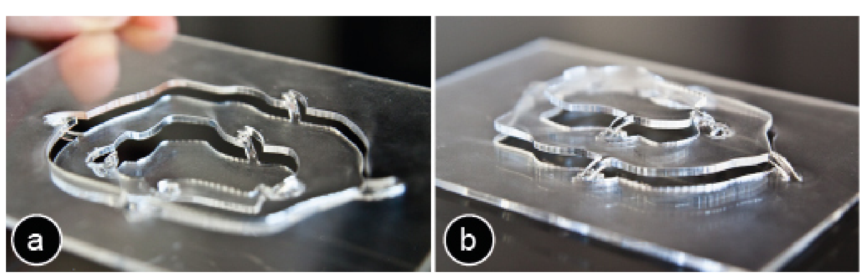

Figure 63: Suspending recursively: If an upper level gets suspended, it moves all lower levels with it. Here we fabricate a multi-level terrain.

Creating a suspender creates a hole in the workpiece—one of the limitations of our approach. However, the approach offers freedom in "routing" the suspenders, which allows us to place them so as to minimize interference with the remaining workpiece (Figure 64b).

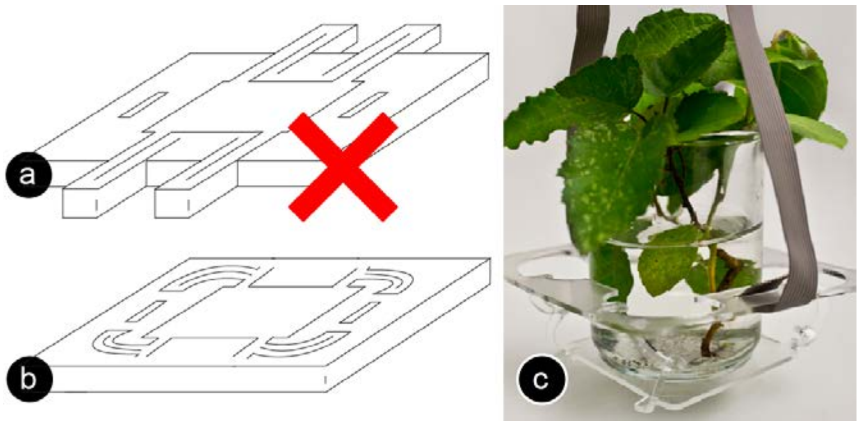

Figure 64: (a) This design does not work, because suspenders break the workpiece apart. (b) Rerouting the suspenders to minimize interference.

4.5.3 *Stretching*

Suspenders are our general mechanism for suspending a patch of material. However, in some cases, the material for the suspenders is required, such as for the paint holder shown in Figure 65b. In this case, we can suspend by stretching. To suspend a patch by stretching, we heat up its outline until it gets compliant and stretches due to the weight of the suspended patch. If a patch is too light, we can add weights to it before suspending (Figure 65a).

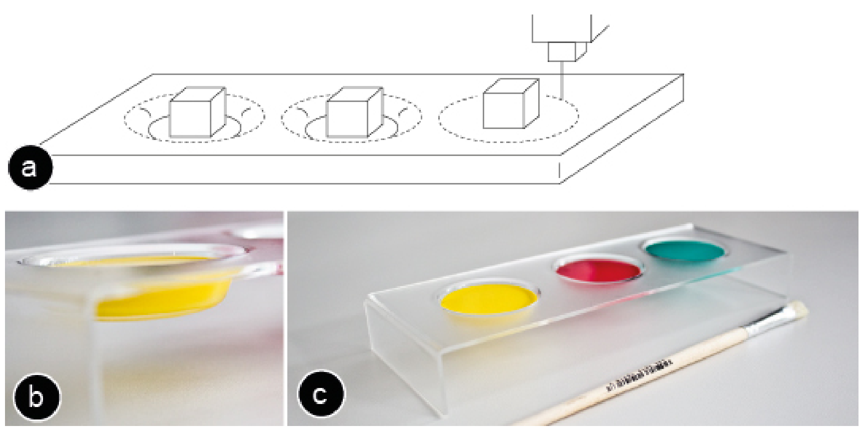

Figure 65: Suspending a patch using stretching creates a container that can hold a liquid.

Stretching causes the walls of the suspended patch to get thinner, which limits the maximum suspension depth. How deep a patch can be suspended by stretching depends on the material thickness as well as the width of the stretched region.

4.6 INTERACTIVE LASERORIGAMI

In addition to the CAD-style interaction, LaserOrigami is fast enough to allow for interactive construction of 3D objects. To explore this, we integrated LaserOrigami into our interactive construction platform constructable [39] (Figure 66).

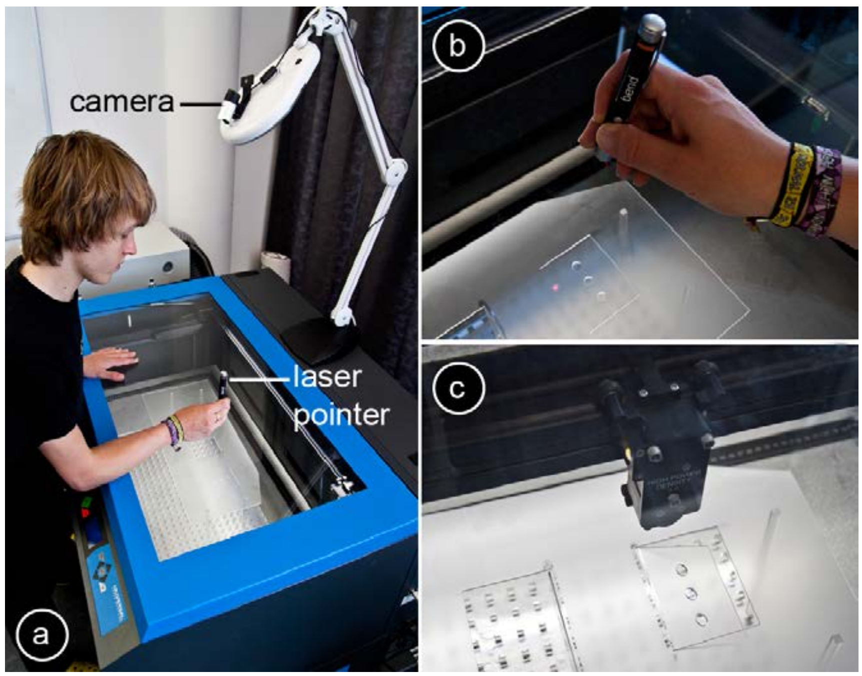

Figure 66: We integrated LaserOrigami into our interactive construction platform constructable: Users interact by drafting directly on the workpiece with hand-held lasers. (b) Here the user draws a bend path using the bend laser. (c) The system responds by bending the selected piece using the defocused laser.

For extending the constructable tool set with the LaserOrigami design elements we turned each of the shape library elements discussed earlier into a separate laser tool, such as the bend, the suspend, and the stretch tool. To set parameters for these tools, such as the size of the bending angle or the length of the suspenders, we added a numpad that allows users to enter a single global parameter that is passed to the current tool.

Figure 67 illustrates how to bend interactively by drawing a stroke across a part of the workpiece using the bend tool. A bend can only occur between two cuts, which allows Laser-Origami to compute the bend as the intersection of the user's draw path and the existing cuts in the workpiece. Users can create multiple bends efficiently by crossing using a single long stroke (Figure 21b).

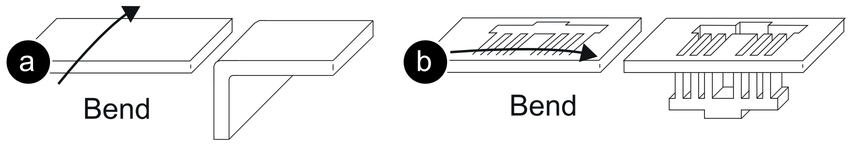

Figure 67: (a) Bending the workpiece by drawing a bend path across. (b) Users can bend multiple parts at once by crossing them all at once.

Interactive LaserOrigami also allows controlling the servomotor by entering the desired angle into the numpad (Figure 68).

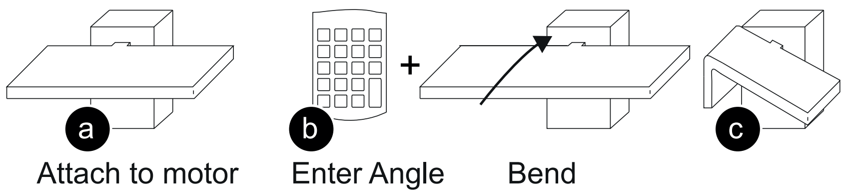

Figure 68: The servomotor allows creating precise bending angles interactively: (a) attach piece to motor, (b) enter bend angle into numpad and draw bend path with *bend* tool, (c) result.

Figure 69 demonstrates how to interactively construct a plant holder using the *suspend* tool, i.e., by (a) drawing the base plate using the *polyline* tool as sketchline, (b) creating the top-level surface using con-

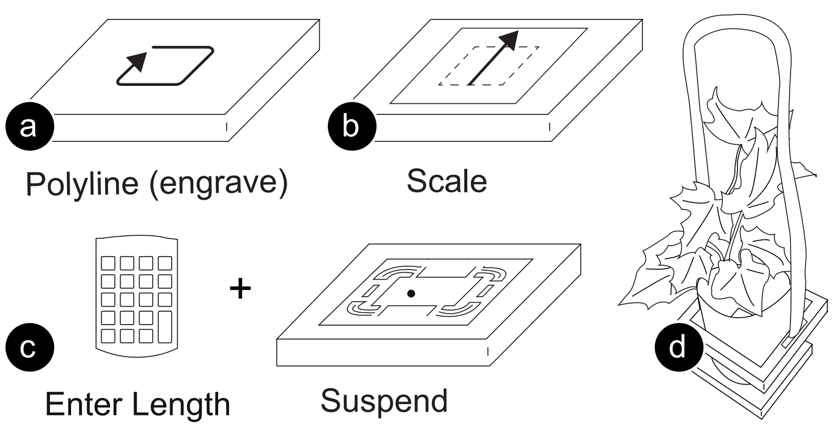

Figure 69: Creating a plant holder interactively.

structable's *scale* tool with depth as a numpad parameter, and (c) suspending the base plate using the *suspend* tool.

4.7 TECHNICAL DETAILS

In this section we provide the details required to replicate the presented work. Master shapes (in combination with the laser cutter's configuration file) encapsulate our knowhow of where to heat and in what order. For our PLS6.150D laser cutter with a 2.0" lens and 1.5mm thick acrylic, for example, we defocus the laser by 50 mm and use 40 percent speed and 30 percent power and we move the laser 40 times across each bend and 6 times across each inflection point of each suspender.

To maximize the time the laser is actually running and minimize the time elements cool off, we create pairs of bend lines of opposite orientation. For the same reason, we perform multiple bend lines on a single suspender before heating up the other suspenders in the same group. To compensate for heat loss along edges, we make bend lines protrude past object outlines.

To make sure all elements unfold properly, we continue to heat all inflection points past the moment when an object is starting to bend. In the case of suspenders, the inflection point in the center moves not only vertically, but also horizontally while dropping (see Figure 60). We compensate for this horizontal movement by creating additional bend lines located across the different horizontal positions of the dropping inflection point.

The PLS6.150D laser points down at an angle, which introduces a horizontal offset during defocusing. We compensate for the offset by shifting bend lines accordingly.

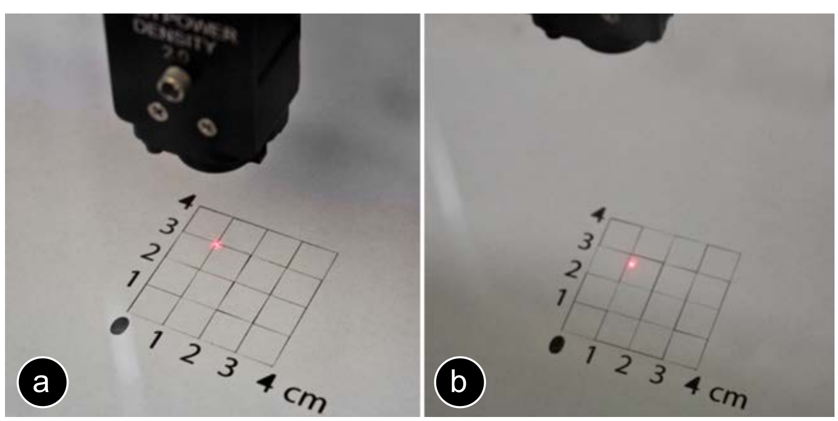

Figure 70: Offset: (a) focused laser, (b) defocused laser.

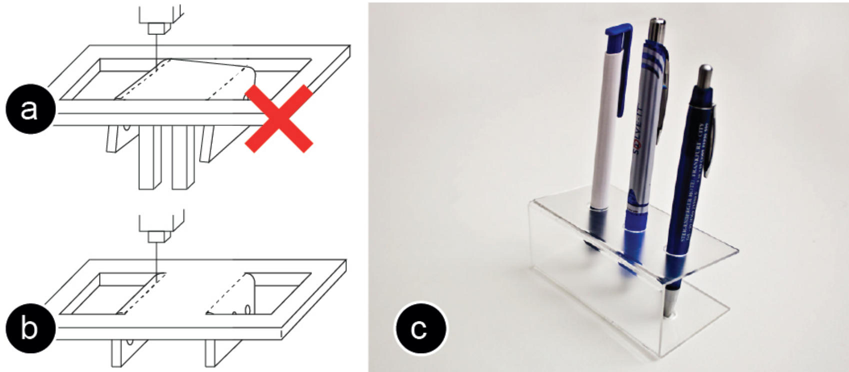

Figure 71: (a) Cutting the entire outline of the workpiece causes it to wiggle, which renders subsequent bending steps imprecise. (b) To keep the workpiece stable, we keep it attached to the main sheet until all bending is done.

To maintain stability during cutting, we keep the workpiece connected to the main acrylic sheet until all bending is done (Figure 71).

During bending, the diagonal of the bent piece has to pass the opening. As shown in Figure 72a, this can cause the bent piece to get stuck in the adjacent material, especially if the bent object is small and thus light or if it has an irregular outline. We address this by first cutting out an extended outline, which creates additional space around the workpiece as illustrated by Figure 72b.

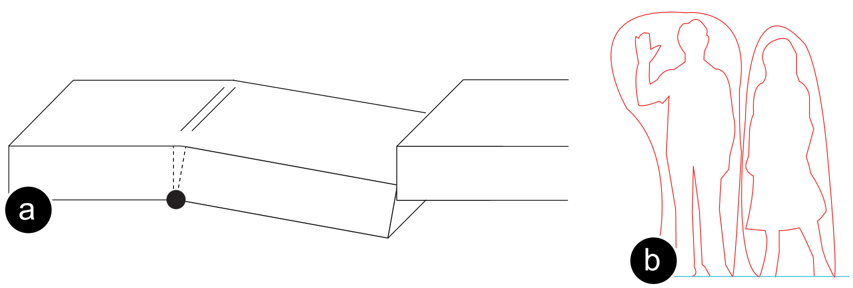

Figure 72: (a) Suspenders of equal length suspend a horizontal patch, (b) unequal suspenders create a ramp.

4.8 EVALUATION: LIVE DEMO AT CHI'13

We gave a demo of LaserOrigami at CHI'13 in Paris over the course of two days. For the demo, we prepared a new object as a give-away for attendees: a small souvenir in the shape of the Paris city outline with the CHI logo attached (see Figure 73). The Paris city outline was 12 cm wide and 7 cm high, and contained a single bend to fold the base for upright standing on a flat surface.

Figure 73: The example object we fabricated at the CHI'13 demo: a give-away in the form of the Paris city outline.

Figure 74 shows the underlaying vector file created in Adobe Illustrator and the settings for the laser cutter we used for each color. The overall fabrication time was 3.10 minutes.

As can be seen in Figure 74, we used an additional outline around the part that had to bend down. The reason for this is that the patch was not heavy enough to bend down under gravity. The additional outline added material, and thus weight to the patch. Since the outline was only loosely attached to the patch, it was easy to break off after all laser cutting and bending was done.

Over the course of two days, we fabricated more than fourty give-aways. While we did not measure the bending angle for each give-away, they all stood up-right in a stable position and had an approximate 90 degree angle. This is in accordance with our earlier tests for

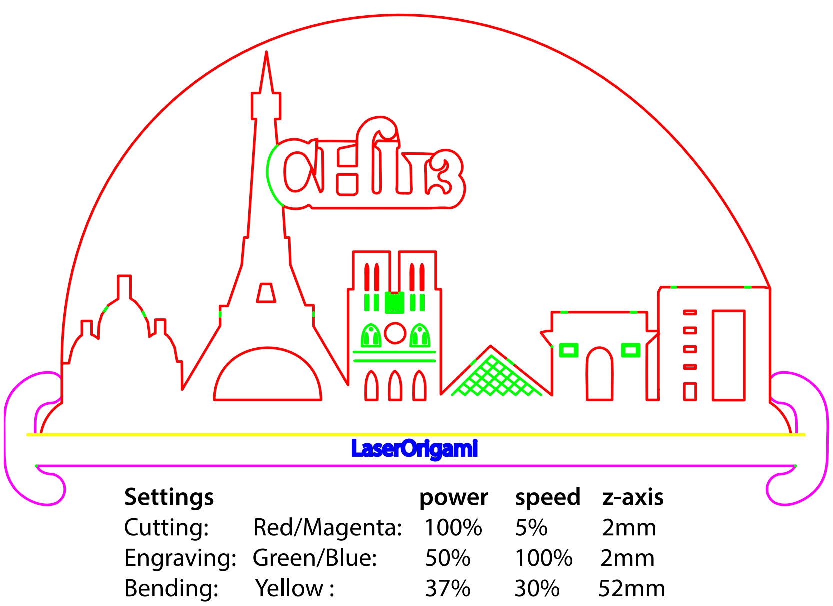

Figure 74: Vector file with color-coded laser cutter settings.

the more complex suspenders, which also had only minor variations in the millimeter range (Figure 75).

Figure 75: Suspenders fabricated in one row show only minor variations.

4.9 FUTURE WORK: ITERATIVE BENDING USING A HEAT CAMERA

LaserOrigami is the first rapid prototyping system that produces physical 3D objects using a laser cutter by bending rather than joining. Based on this mechanism, LaserOrigami eliminates the need for manual assembly since it produces 3D folded objects in a single integrated process.

As future work, we plan to eliminate the need for pre-programming laser cutter behavior by using a regular webcam and a heat camera. The idea is to use the heat camera to monitor the temperature of the acrylic at the bend path and to dynamically adjust the laser-cutter power depending on the current temperature of the region. The best temperature for bending is around 130 degree for 1.5mm acrylic, temperatures below that point will not lead to compliance, temperatures above that point lead to bubbles in the acrylic, which are aesthetically unpleasant. For this, we are currently experimenting with the heat imaging array Melexis MLX90620 [5] that is shown in Figure 76. Finally, we plan to add a regular webcam to LaserOrigami to observe the shape of the workpiece, i.e. to stop the heating process once the desired bending angle is achieved.

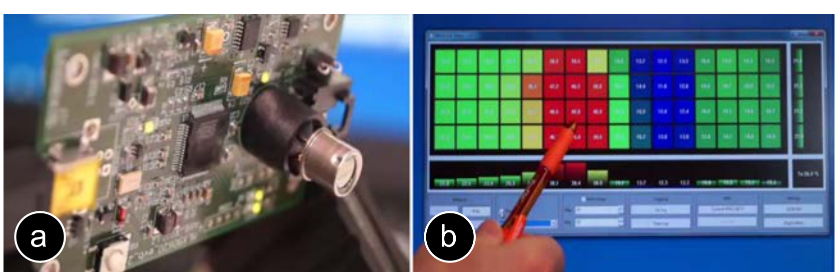

Figure 76: A heat cam can be used to dynamically adjust the laser's power.

5 CONCLUSION AND FUTURE WORK

5.1 INTERACTIVE CONSTRUCTION

The main objective of this thesis was to propose the concept of interactive construction as a way to enable users to create functional mechanical devices with an interactive fabrication system. To accomplish this, we re-introduced the benefits of CAD: precision, trial and error, and fast interaction, back into the interactive fabrication process, as those qualities got largely lost when the interactive paradigm changed towards more direct interaction on the workpiece.

We implemented a system called constructable that demonstrates how the CAD qualities can be integrated into the interactive fabrication paradigm. constructable achieves precision by building constraints into each tool, by providing sketch lines as an alignment aid, and by using the laser cutter itself for high precision output after each editing step. Finally, constructable allows for trial and error through either a joint based repair or through refabrication. Finally, constructable allows for fast interaction because it uses a high-powered laser-cutter to create the physical output after each editing step. Since our second interactive construction system LaserOrigami is built on top of constructable, it provides the same three benefits. In addition, LaserOrigami allows for the interactive construction of three dimensional objects without any manual assembly by the user.

Interactive construction combines the benefits from both worlds: the direct interaction of interactive fabrication with the precision and undo from CAD. This makes construction tasks more accessible for users that are not experts in digital editing: Drawing with a laser-pointer through the safety glass enclosure is a task comparable to drawing with a pen on a piece of paper.

We think that interactive construction is best used for quickly prototyping simple objects, such as the booklet from Figure 77a. We stretched the concept of interactive construction to show what is possible when creating the slightly more advanced gearbox from Figure 77b. However, for projects that require a lot of planning ahead, a digital editor might still be the tool of choice.

Our interactive construction system constructable and the integrated bending technique LaserOrigami are based on a laser cutter since laser cutters are the fastest fabrication devices to date. The reason for

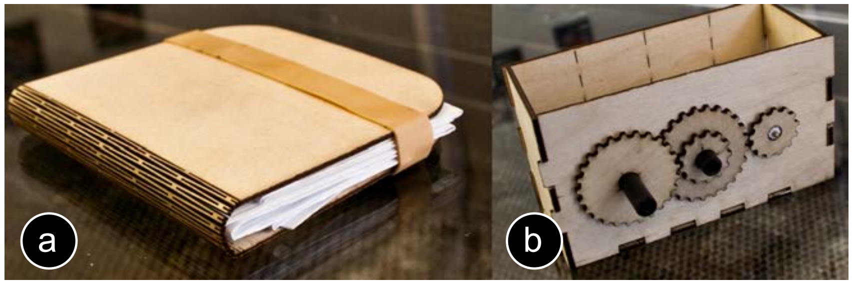

Figure 77: (a) We think most use cases for interactive construction will be simple, such as this wooden booklet. (b) This gearbox is possible with interaction construction, but might be better done in a 3D editor.

this is that laser cutters only cut lines, a 2D operation, which speeds up the process tremendously compared to 3D fabrication with milling machines and 3D printers that operate in three dimensions.

Recently, a new additive fabrication device in the shape of a hand-held pen called the 3Doodler [2] was developed that allows users to draw lines into mid-air (Figure 78). It works by feeding a plastic string from the back of the device through the heated pen nozzle. When the plastic leaves the nozzle it is soft and then cools down quickly to solidify into the shape currently drawn by the user. Since the 3Doodler is made for wire-like drawings based on 2D lines, it is also very fast in creating shapes.

As with previous interactive fabrication systems the 3Doodler allows only for freehand drawing and does not offer any support for the user to achieve a precise construction task. In Figure 78, the user is trying to draw a cube, but all its sides are jerky and have different lengths. We think the 3Doodler provides a good basis to extend interactive construction towards additive fabrication.

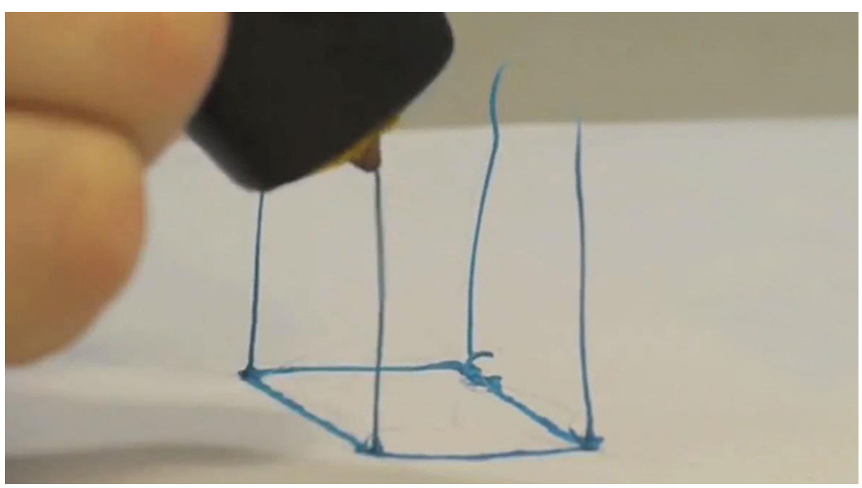

Figure 78: 3Doodler, freehand 3D drawing in mid-air.

5.2 PERSONAL COMPUTING AND PERSONAL FABRICATION

We want to conclude this thesis by rehashing our analogy between personal computing and personal fabrication that we made in the introduction.

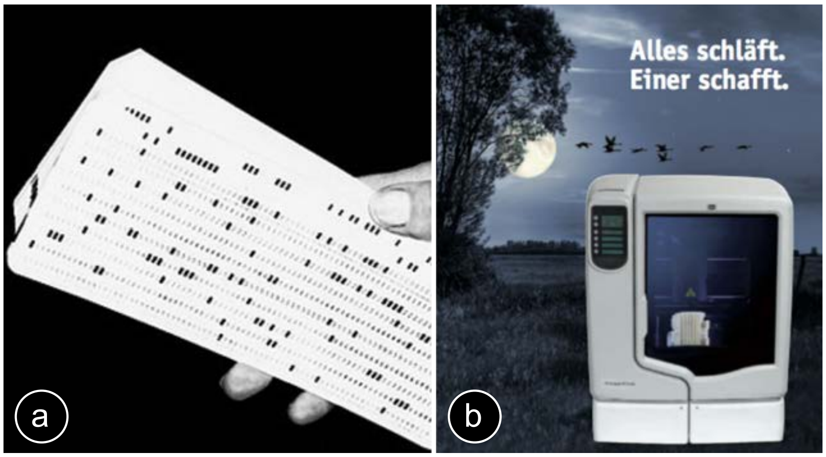

Figure 79: Slow hardware results in delayed feedback, which is especially frustrating for beginners, who repeatedly make mistakes.

In the same way as personal computing did not spread until more accessible interfaces were developed, personal fabrication will not spread before we provide interfaces that make them more easy to use and provide a better learning curve for novice users. Personal computing showed that the main breakthrough happened when the interaction paradigm changed from overnight-processing with delayed feedback towards interactive computing with immediate feedback after every editing step. However, this change was only possible when faster hardware became available that was able to process the input events in real-time.

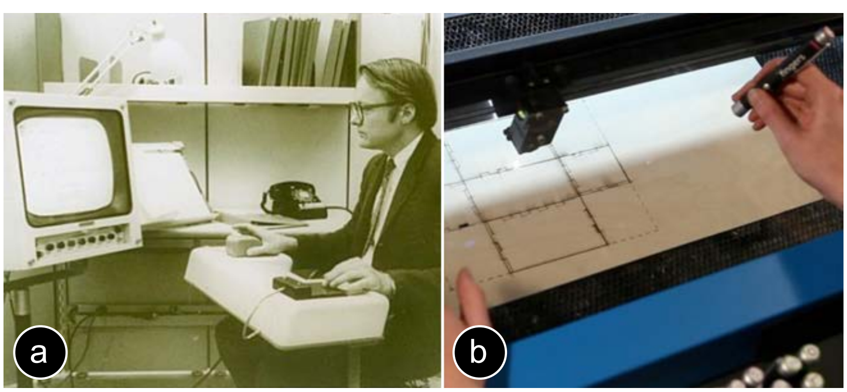

Figure 80: Faster hardware allows for immediate feedback and thus a good learning curve for novices.

With personal fabrication, we are still at the stage of overnight printing with delayed feedback. This forces users to work through the indirection of a CAD system, although working directly on the physical matter with a hand-held tool is more intuitive as users already know it from traditional crafting. The main hurdle to overcome here is the same as in personal computing: slow hardware, such as 3D printers, prevent us from interactive editing, as they need hours to create the physical object.

To spread personal fabrication, we need two things: we need to provide more accessible interfaces, and to make this possible, we need faster hardware.

BIBLIOGRAPHY

[1] 3D printer, D. s. e. http://www.stratasys.com/3d-printers/design-series/performance/dimension-1200es.

[2] 3Doodler. http://www.the3doodler.com/.

[3] Apitz, G., and Guimbretière, F. Crossy: A crossing-based drawing application. In *Proceedings of UIST* (2004), pp. 3–12.

[4] Arisandi, R., Takami, Y., Otsuki, M., Kimura, A., Shibata, F., and Tamura, H. Enjoying virtual handcrafting with tooldevice. In *Adjunct Proceedings of UIST* (2012), pp. 17–18.

[5] Array, M. I. http://www.melexis.com/Infrared-Thermometer-Sensors/Infrared-Thermometer-Sensors/MLX90620-776.aspx.

[6] Balkcom, D. J., and Mason, M. T. Introducing robotic origami folding. In *Proceedings of ICRA* (2004), pp. 3245–3250.

[7] Bartkowiak, K., Edwardson, S., Borowski, J., Dearden, G., and Watkins, K. Laser forming of thin metal components for 2d and 3d applications using a high beam quality, low power nd:yag laser and rapid scanning optics. In *International Workshop on Thermal Forming* (2005).

[8] Bier, E. Snap dragging. In *Proceedings of SIGGRAPH* (1986), pp. 233–240.

[9] Brockmeyer, E., Poupyrev, I., and Hudson, S. Papillon: Designing curved display surfaces with printed optics. In *Proceedings of UIST* (2013), pp. 457–462.

[10] Brown, C., and Hurst, A. Viztouch: Automatically generated tactile visualizations of coordinate spaces. In *Proceedings of TEI* (2012), pp. 131–138.

[11] Buxton, B. Sketching user experiences: Getting the design right and the right design. In *Morgan Kaufmann* (2007).

[12] Buxton, W. A three-state model of graphical input. In *Proceedings of INTERACT* (1990), pp. 449–456.

[13] CHI 2013 Conference, Paris, F. http://chi2013.acm.org/.

[14] Csikszentmihalyi, M. Flow: The psychology of optimal experience. harper perennial modern classics.

[15] Deng, D., and Chen, Y. Design of origami sheets for foldable object fabrication. In *Proceedings of ASME* (2012).

[16] Douglas Engelbart, The Mother of All Demos, . http://www.youtube.com/watch?v=JfIgzSoTMOs.

[17] Fitzmaurice, G., Ishii, H., and Buxton, B. Bricks: Laying the foundations for graspable user interfaces. In *Proceedings of CHI* (1995), pp. 442–449.

[18] Fitzmaurice, G., Khan, A., Pieke, R., Buxton, B., and Kurtenbach, G. Tracking menus. In *Proceedings of UIST* (2003), pp. 71–79.

[19] Follmer, S., Carr, D., Lovell, E., and Ishii, H. Copycad: remixing physical objects with copy and paste from the real world. In *Adjunct Proceedings of UIST* (2010), pp. 381–382.

[20] Fukuchi, K., and Jo, K. Lasercooking: An automated cooking technique using laser cutter. In *Adjunct Proceedings of TEI* (2012), pp. 163–168.

[21] Geiger, M., and Vollertsen, F. The mechanisms of laser forming. In *CIRP ANNALS* (1993), pp. 301–304.

[22] Gershenfeld, N. Fab: The coming revolution on your desktop–from personal computers to personal fabrication. In *Basic Books* (2007).

[23] Hinckley, K., Baudisch, P., Ramos, G., and Guimbretiere, F. Design and analysis of delimiters for selection-action pen input phrases in scriboli. In *Proceedings of CHI* (1996), pp. 451–460.

[24] Hinckley, K., Guimbretiere, F., Baudisch, P., Sarin, R., Agrawala, M., and Cutrell, E. The springboard: Multiple modes in one spring-loaded control. In *Proceedings of CHI* (2006), pp. 181–190.

[25] Jansen, Y., Dragicevic, P., and Fekete, J.-D. Evaluating the efficiency of physical visualizations. In *Proceedings of CHI* (2013), pp. 2593–2602.

[26] Lang, R., and Mcey, R. The complete book of origami: Step-by step instructions in over 1000 diagrams.

[27] Laser cutter LS3020, H. L. L. http://hpclaser.co.uk.

[28] Laser cutter PLS6.150D, U. L. S. http://www.ulsinc.com/products/pls6150d/.

[29] Lasersaur. http://labs.nortd.com/lasersaur.

[30] Lau, M., Saul, G., Mitani, J., and Igarashi, T. Modeling-in-context: user design of complementary objects with a single photo. In *Proceedings of SBIM* (2010), pp. 17–24.

[31] Lee, J. Haptic intelligentsia. http://studio-homunculus.com/portfolio/haptic-intelligentsia-human-prototyping-machine.

[32] Li, W., and Yao, Y. L. Laser bending of tubes: Mechanism, analysis, and prediction. *ASME Journal of Manufacturing Science and Engineering 123*, 4 (2001), 674–681.

[33] Li, W., and Yao, Y. L. Numerical and experimental investigation of convex laser forming process. *SME Journal of Manufacturing Processes 3*, 2 (2001), 73–81.

[34] Liao, C., Guimbretière, F., and Hinckley, K. Papiercraft: a command system for interactive paper. In *Proceedings of UIST* (2005), pp. 241–244.

[35] Liu, Y., Boyles, J., Genzer, J., and Dickey, M. Self-folding of polymer sheets using local light absorption. *Journal of Soft Matter 8*, 6 (2012), 1764–1769.

[36] Makerbot. http://www.makerbot.com/.

[37] Mori, Y., and Igarashi, T. Plushie: An interactive design system for plush toys.

[38] Mueller, S., Kruck, B., and Baudisch, P. Laserorigami: Laser-cutting 3d objects. In *Proceedings of CHI* (2013), pp. 2585–2592.

[39] Mueller, S., Lopes, P., and Baudisch, P. Interactive construction: Interactive fabrication of functional mechanical devices. In *Proceedings of UIST* (2012), pp. 599–606.

[40] Myers, B., Bhatnagar, R., Nichols, J., Peck, C., Kong, D., Miller, R., and Long, A. Interacting at a distance: Measuring the performance of laser pointers and other devices. In *Proceedings of CHI* (2002), pp. 33–40.

[41] Ogawa, H., Mara, M., Lindinger, C., Gardiner, M., Haring, R., Stolarsky, D., Ogawa, E., and Hörtner, H. Shadowgram: A case study for social fabrication through interactive fabrication in public spaces. In *Proceedings of TEI* (2012), pp. 57–60.

[42] Oh, Y., Johnson, G., Gross, M., and Do, E. The designosaur and the furniture factory. *Proceedings of DCC* (2006), 123–140.

[43] Paulson, B., and Hammond, T. A system for recognizing and beautifying low-level sketch shapes using ndde and dcr. In *Adjunct Proceedings of UIST* (2007).

[44] Posch, I., and Fitzpatrick, G. First steps in the fablab: Experiences engaging children. In *Proceedings of OzCHI* (2012), pp. 497–500.

[45] Qi, J., and Buechley, L. Animating paper using shape memory alloys. In *Proceedings of CHI* (2010), pp. 749–752.

[46] Qi, J., and Buechley, L. Electronic popables: Exploring paper-based computing through an interactive pop-up book. In *Proceedings of TEI* (2010), pp. 121–128.

[47] Rivers, A., Moyer, I., and Durand, F. Position-correcting tools for 2d digital fabrication. *ACM Transactions on Graphics (TOC) 31*, 4 (2012).

[48] Saul, G., Lau, M., Mitani, J., and Igarashi, T. Sketchchair: An all-in-one chair design system for end users. In *Proceedings of TEI* (2011), pp. 73–80.

[49] Saul, G., Xu, C., and Gross, M. D. Interactive paper devices: End-user design and fabrication. In *Proceedings of TEI* (2010), pp. 205–212.

[50] Savage, V., Chang, C., and Hartmann, B. Sauron: Embedded single-camera sensing of printed physical user interfaces. In *Proceedings of UIST* (2013), pp. 447–456.

[51] Savage, V., Zhang, X., and Hartmann, B. Midas: Fabricating custom capacitive touch sensors to prototype interactive objects. 579–588.

[52] Shapeoko. http://www.shapeoko.com/.

[53] Song, H., Guimbretière, F., Hu, C., and Lipson, H. Modelcraft: Capturing freehand annotations and edits on physical 3d models. In *Proceedings of UIST* (2006), pp. 13–22.

[54] Source, L. L. O. http://www.laoslaser.org/.

[55] Tanenbaum, J., Williams, A., Desjardins, A., and Tanenbaum, K. Democratizing technology: Pleasure, utility and expressiveness in diy and maker practice. In *Proceedings of CHI* (2013), pp. 2603–2612.

[56] TEI 2013 Conference, Barcelona, S. http://www.tei-conf.org/13/home.

[57] Whitney, J., Sreetharan, P., Ma, K., and Wood, R. Pop-up book mems. *Journal of Micromechanics and Microengineering 21*, 11 (2012).

[58] Wibowo, A., Sakamoto, D., Mitani, J., and Igarashi, T. Dressup: A 3d interface for clothing design with a physical mannequin. In *Proceedings of TEI* (2012), pp. 99–102.

[59] Willis, K., Brockmeyer, E., Hudson, S., and Poupyrev, I. Printed optics: 3d printing of embedded optical elements for interactive devices. In *Proceedings of UIST* (2012), pp. 589–598.

[60] Willis, K., Lin, J., Mitani, J., and Igarashi, T. Spatial sketch: Bridging between movement & fabrication. In *Proceedings of TEI* (2010), pp. 5–12.

[61] Willis, K., Xu, C., Wu, K., Levin, G., and Gross, M. Interactive fabrication: New interfaces for digital fabrication. In *Proceedings of TEI* (2011), pp. 69–72.

[62] Wohlers Associates, Unit Sales Estimates of Personal 3D Printers, I. B. . http://wohlersassociates.com/brief07-12.htm.

[63] Yahoo-Research-Japan. Tangible search: A voice-activated search engine and a 3d printer brings the internet to blind kids.

[64] Zoran, A., and Paradiso, J. Freed – a freehand digital sculpting tool. In *Proceedings of CHI* (2012), pp. 2613–2616.

[65] Zoran, A., Shilkrot, R., and Paradiso, J. Human-computer interaction for hybrid carving. In *Proceedings of UIST* (2013), pp. 433–440.

DECLARATION

I certify that the material contained in this thesis is my own work and does not contain unreferenced or unacknowledged material. I also warrant that the above statement applies to the implementation of the project.

Hiermit versichere ich, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Hilfsmittel verwendet habe. Ich erkläre hiermit weiterhin die Gültigkeit dieser Aussage für die Implementierung des Projekts.

Potsdam, October 2013

Stefanie Mueller