JigFab: Computational Fabrication of Constraints to Facilitate Woodworking with Power Tools

Danny Leen[2], **Tom Veuskens**[3], **Kris Luyten**[1], **Raf Ramakers**[1]
[1]Hasselt University - tUL - Flanders Make, EDM, Belgium, firstname.lastname@uhasselt.be
[2]LUCA School of Arts, KULeuven, Belgium, danny.leen@luca-arts.be
[3]Hasselt University - tUL - Flanders Make, EDM, Belgium, firstname.lastname@student.uhasselt.be

ABSTRACT

We present JigFab, an integrated end-to-end system that supports casual makers in designing and fabricating constructions with power tools. Starting from a digital version of the construction, JigFab achieves this by generating various types of constraints that configure and physically aid the movement of a power tool. Constraints are generated for every operation and are custom to the work piece. Constraints are laser cut and assembled together with predefined parts to reduce waste. JigFab's constraints are used according to an interactive step-by-step manual. JigFab internalizes all the required domain knowledge for designing and building intricate structures, consisting of various types of finger joints, tenon & mortise joints, grooves, and dowels. Building such structures is normally reserved for artisans or automated with advanced CNC machinery.

KEYWORDS

Computational Fabrication, Wood Working, CAD

ACM Reference format:
Danny Leen[2], Tom Veuskens[3], Kris Luyten[1], Raf Ramakers[1]. 2019. JigFab: Computational Fabrication of Constraints to Facilitate Woodworking with Power Tools. In *Proceedings of ACM Conference on Human Factors in Computing Systems, Glasgow, UK, May 4–9, 2019 (CHI'19),* 12 pages.
https://doi.org/10.1145/3290605.3300386

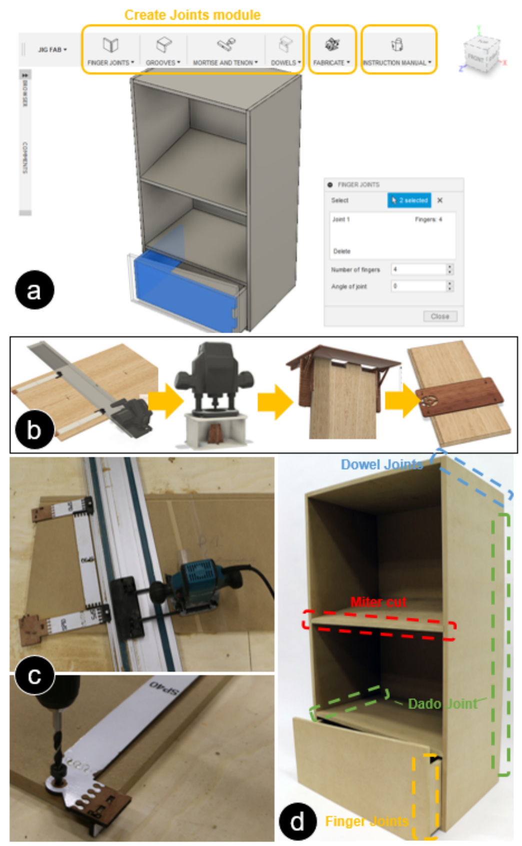

Figure 1: JigFab's end-to-end design and fabrication workflow: (a) Embedding joints in the 3D model, (b) Custom constraints and instructions are generated, (c) Fabrication by constraining power tools, (d) The assembled work piece.

1 INTRODUCTION

Personal fabrication tools, such as desktop 3D printers and laser cutters enable people without crafting experience to produce physical artifacts. While these types of machines are ideal for making desktop scale artifacts, producing larger than desktop size models requires multi-axis CNC machines or robotic arms capable of 3D printing which are expensive, bulky, and hard to setup and operate. Hence constructions, such as cabinets, tables, and houses are still made using traditional power tools that operate locally, on a specific part of the workpiece. Making precise constructions with power tools, however, requires significant expertise only available to artisans.

CHI'19, May 4–9, 2019, Glasgow, UK

ACM ISBN 978-1-4503-5970-2/19/05...$15.00
https://doi.org/10.1145/3290605.3300386

Previous research enabled people with limited crafting experience to make large objects by manufacturing only complex joints with digital fabrication machinery and using existing parts for the remaining construction, such as lumber [19], bottles [10], or LEGO bricks [21]. While suitable for truss-like objects, this substitution approach often impacts the fidelity for objects with more complex shapes [21]. Researchers also explored orthogonal approaches to facilitate building large constructions by integrating actuators in handheld tools, such as routers [26, 32, 33], that partially automate the complex operations. As routers are not sufficient for every operation in a construction project, the Drill Sergeant system [28] augments a wide variety of power tools with electronic sensors. Such tools however do not assist users in organizing all operations to go from raw material to an intricate construction consisting of many parts and complex joinery.

In contrast to these approaches, professional craftsmen build custom jigs or use reconfigurable jigs and fixtures to constrain the position, movement, and angle of the power tool with respect to the workpiece. Examples include drill guides, pocket jigs, and box joint jigs for making finger joints. Power tools operating in these jigs are guided along a path and ensure that the precision of the power tool does not solely rely on the steadiness of the user's hands. When properly configured, jigs and fixtures support in making highly sophisticated constructions that match the required specifications and can be replicated using the same process and constraints. In this work, we present a novel system that uses constraints for every operation performed on the workpiece, from constraining movements and positions to configuring the orientation and cut/drill depth of power tools. While traditional jigs and fixtures are very complex to configure and therefore targeted towards professionals, our constraints are designed and generated to fit the specifics of a digital version of the work piece and thus do not require fine-tuning. This approach allows for precise manufacture of objects without requiring users to measure, which is the source of the many errors in fabrication work [8].

In this paper, we present *JigFab*, an integrated end-to-end system that enables casual makers to design and fabricate constructions with power tools. Starting from a digital version of the design (Figure 1a), JigFab achieves this by generating custom constraints that precisely configure power tools on top of a work piece. JigFab supports the use of various power tools, including a circular saw, a plunge router, a drill, and a compound miter saw to fit the operation at hand (Figure 6). While small constraints are produced with a CO_2 laser cutter, large constraints mainly consist of predefined elements to keep waste to a minimum (white parts in Figure 1c). JigFab provides an extensive interactive step-by-step manual that assists in the assembly of constraints and their alignment on the workpiece and power tool (Figure 1b). While the concept of computationally generated jigs apply to many types of materials, the prototypes presented in this paper are made of wood. JigFab therefore supports, and assists the user in designing parametric finger joints, dowel joints, miter joints, grooves, and mortise and tenon joints for a total of 16 different joints, according to the classification of wood joints as described by Jackson and Day [6]. Our main contribution is an end-to-end system and corresponding workflow that facilitates the design of advanced constructions and assist users in fabricating these constructions with power tools. Specifically, we contribute:

(1) A construction support set of parametric jigs, fixtures, and templates for shaping parts of the work piece and making all supported joints using a variety of power tools, including a circular saw, a plunge router, a hand-held drill, and a compound miter saw to fit the operation at hand.

(2) A software environment that computationally generates custom constraints from a digital version of a construction, consisting of parametric joints. Our technique generates constraints to facilitate every operation, from slicing parts to fabricating intricate joints.

2 RELATED WORK

This work draws from, and builds upon prior work in Computer-Aided Design for carpentry and furniture design, smart crafting tools, and large-scale fabrication.

CAD for Carpentry and Furniture Design

CAD environments for furniture design, such as SketchList3D [29] support professional carpenters in designing joints. With the current popularity of digital fabrication, new CAD environments have been developed to automate the design of 3D printed or laser cut joints. This resulted in various box joint generators that are available as online generators [20, 39] and as plugins in vector graphics tools [23]. In a similar vein, Magrisso et al. [19] explored generative design approaches for joining wood with 3D printed connectors.

Beyond the specifics of fabrication machinery, researchers also investigated computational methods to facilitate furniture design, assembled solely with intricate joints. Fu et al. [4, 31] presented a system for solving a network of interlocking joints to build furniture held together with a single key serving as a global interlocking mechanism. Instead of fully automating the design and positioning of joints, Yao et al. [38] introduced an interactive system that empowers users to sketch free-form aesthetic joints after which the system generates solid 3D parts and analyzes its stability.

When feedback on the stability of the entire furniture design, subjected to external forces, is desired, the system of Umetani et al. [35] offers suggestions to improve the positioning of parts.

In addition to facilitate the design of traditional wood joinery, researchers built systems to computationally optimize furniture designs for easy assembly using connectors, such as metal frames, hinges, and bolts. In this context, Lau et al. [13] build a formal grammar from IKEA parts and connectors to automate the positioning of connectors in new designs. A similar system was developed by Schulz et al. [14] which additionally supports users in exploring design alternatives for furniture design. Using the system of Yang [37], a furniture model can be optimized to match the specific qualities of the material used during the fabrication process. To facilitate furniture design further, sketching interfaces have been developed for furniture design [15, 22]. These systems optimize the regularity of the object and apply extrusions and basic joints to fabricate the object using 3D printing or laser cutting. SketchChair [27] takes a similar approach but specifically focuses on the design and stability of chairs.

JigFab similarly aids the design of 16 types of woodworking joints and additionally contributes computational methods to automate the design of constraints that configure and physically aid the movement of power tools. In earlier research, computational fabrication of physical constraints is rather fragmented and received limited attention. The boxes.py generator [2], for example, supports fixtures for positioning the work piece in slanted configurations in a laser cutter to allow for angular cuts. A few systems also automate the design of attachments to existing objects [3] and between 3D objects [11]. Finally, ProxyPrint [34] computationally generates fixtures to facilitate the fabrication of wireframe objects.

Smart Crafting Tools

There are a number of research projects that augment traditional crafting tools with actuation and correction mechanisms to facilitate [25, 41] the operation of hand-held crafting tools or make them safer in use [40]. Similarly, CNC features have been embedded in power tools, such as routers to simplify their operation for people with limited crafting skills. Examples include Shaper [26], Handibot [33], and MatchSticks [32].

As routers are not sufficient for every operation in a construction project, the Drill Sergeant system [28] enriches a wider variety of power tools. This project does not support the end-to-end process offered by JigFab to move from a 3D version of a construction to its fabrication. The Smart Makerspace [9] guides users step-by-step while building electronic prototypes but only supports predefined tutorials.

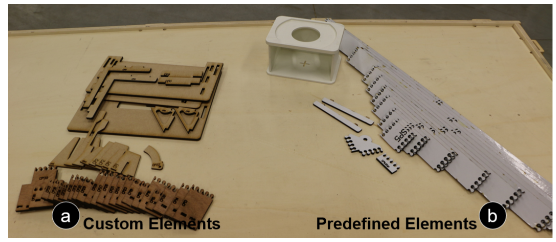

Figure 2: Computationally generated constraints consist of (a) custom laser cut elements and (b) predefined elements.

Lipton et al. [18] presents an end-to-end design and fabrication workflow to automate carpentry but requires an advanced installation with several robots.

Finally, our approach allows for fabricating objects by aligning various constraints and therefore it takes inspiration from research focusing on fabrication workflows that do not require users to measure, such as StrutModeling [16] and MixFab [36].

Large-Scale Digital Fabrication

Previous efforts to facilitate or speed-up digital fabrication of large constructions mainly aimed to replace large elements of the construction with existing parts, such as lumber [19], bottles [10], LEGO bricks [21], or laser cut parts [30]. Alternatively, the entire construction can be fabricated in a low-fidelity version [1, 24] or distributed among robots and many workers [12]. In contrast, JigFab preserves the fidelity of the construction and uses a substitution approach during the generation of constraints, consisting of predefined and custom laser cut elements.

3 THE JIGFAB SYSTEM

Definitions and Overview

The JigFab system generates custom *constraints* to precisely position and operate power tools with respect to the work piece. We define *jigs* as constraints that limit the movement of a power tool along a tool path. *Fixtures* fixate the position and orientation of the work piece with respect to the power tool or jig. Finally, *templates* configure settings of power tools (e.g. drilling/milling depth or cutting angle). The term *constraint* is used, in the remainder of this paper, when referring to all three types.

Every type of constraint consist of *predefined elements*, *custom elements*, and *adapters*. All types are used in combination to keep waste, caused by constraints, to a minimum. *Predefined elements* have a fixed size and are used frequently in JigFab across many types of constraints and projects. Therefore these elements are manufactured beforehand using a laser cutter and their production could even be outsourced.

As shown in Figure 2b predefined elements are the largest elements and have a white color throughout this paper. *Custom elements* are specific to a work piece and oftentimes augment predefined elements to turn them into custom constraints that exactly fit the specifics of a work piece. Figure 2a shows all custom elements required for fabricating the display cabinet with drawer, shown in Figure 1d. While custom elements are designed for one specific operation, JigFab reuses them whenever possible in future operations or projects. Custom constraints are fabricated with a laser cutter and are made of plywood and MDF in this paper. They can be recognized throughout this paper by their brown color and wood texture. Both custom and predefined elements have unique labels engraved to facilitate their identification and assembly. Finally *adapters* are small 3D printed black parts that make commercially available power tools compatible with JigFab's jigs, fixtures, and templates. Similar to predefined elements, adapters are fabricated once and reused across projects.

Walkthrough

The following walkthrough gives a brief overview of JigFab's integrated end-to-end fabrication workflow. The JigFab design environment is implemented as a plugin for Autodesk Fusion 360 [5], a popular and widespread CAD environment. Users start by designing constructions in Fusion's modeling workspace. When switching to the JigFab workspace within Fusion 360, three modules are available: *Create joints*, *Fabricate*, and *Instruction Manual* (Figure 1a). At any time, the user can switch back to the modeling workspace to adapt the design or add new elements. The three modules in the JigFab workspace are used in sequence.

Step 1: Generation of Joints. The *Create joints* module generates joints between two parts and has four buttons, one for every supported type of joint (Figure 1a): finger joints, dowels, grooves, and mortise & tenon joints. Each of these options adds the respective woodworking joint between two selected parts of the work piece. For generating these joints, good practices in woodworking are used, such as the depth of a groove is by default half the thickness of the board. More experienced users can change these default settings to further fine-tune joints. For example, the number of finger joints or the angle to realize dove tail joints. In a similar vein, the groove operation supports offsets in two dimensions to realize a stopped housing joint, a tapered housing joint, or a combination of both. When specifying new joints or changing parameters, JigFab renders a preview of the joint on the user's 3D model in real-time.

Step 2: Generation of Custom Constraints. The *Fabricate* module starts by entering the dimensions of all available material (boards and lumber) for building the work piece. The user also specifies the kerf of the laser cutter, the diameter of the available milling bit, and the thickness of the material for laser cutting the custom constraints. JigFab first solves the cutting stock problem and fits all parts of the work piece on the available items of stock material entered by the user. The user is prompted when there is insufficient stock material and no solution is found.

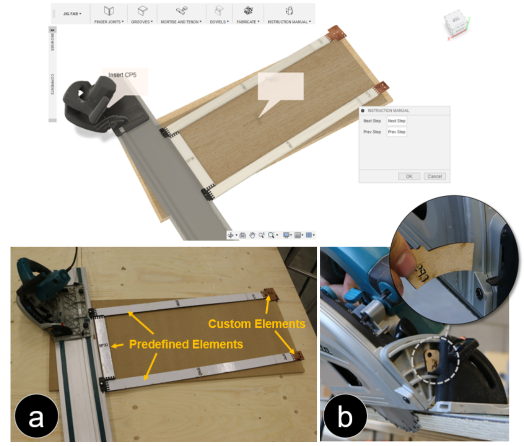

Figure 3: A miter-bevel cut consists of (a) constraints to match the ruler guide with the miter angle, (b) a template to transfer the bevel angle to the miter saw.

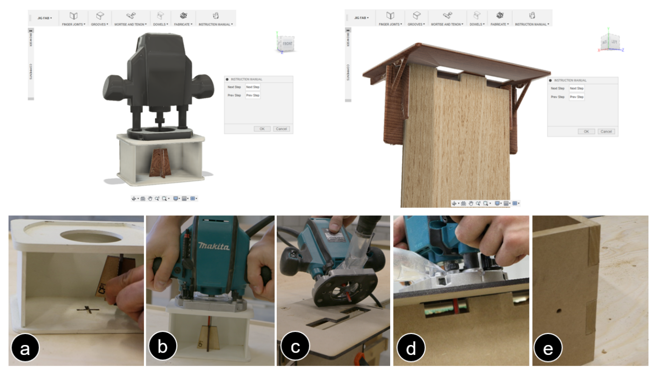

Figure 4: Making a finger joint: (a-b) setting the depth of the router using the depth transfer box and a custom template, (c-e) cutting the fingers using the custom generated finger joint jig

For every slicing operation as well as for every operation required for making the parametric joints, specified in Step 1, JigFab now configures the appropriate constraints that will facilitate the operation of power tools in every step. The *Fabricate* module finishes by exporting DXF files for all custom elements of a constraint, ready for laser cutting. Figure 4a shows an assembled jig for a finger joint consisting of only custom parts while Figure 3a-b shows a fixtures for a

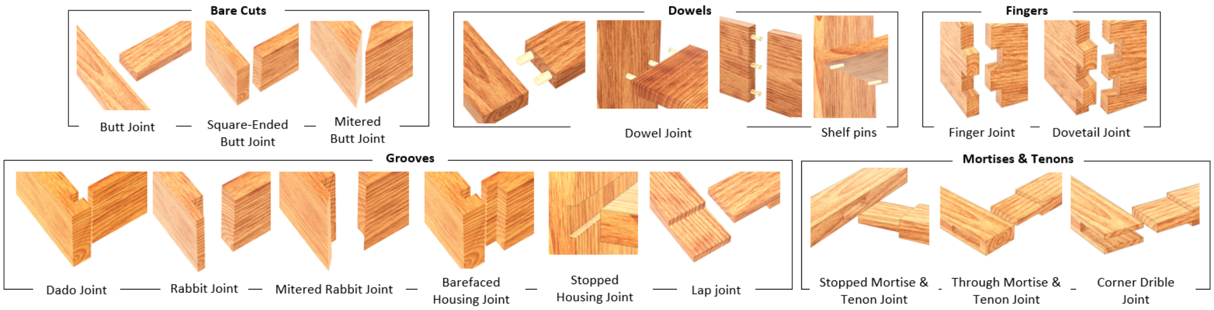

Figure 5: JigFab supports 16 joints categorized in groups

bevel-miter cut consisting of mainly predefined parts (white parts).

Step 3: Slicing with Power Tools. The *Instruction manual* module consists of a series of interactive step-by-step instructions that users navigate through at their own pace. It starts with a series of slicing operations. As shown in Figure 3, every instruction is an interactive 3D rendering and shows which custom or predefined elements to use, the assembly of the constraints, the appropriate power tool to use, and the positioning of constraints on the work piece or power tool. Figure 3 shows an example instruction for making a bevel cut with a circular saw: (a) First, predefined and custom puzzle-shaped elements are aligned to fixate the work piece with respect to the power tool. (b) A custom laser-cut template is inserted in the power tool to configure its bevel angle. After making the cut, the user labels the new parts as instructed by writing down the part number and two arrows that define its orientation. Similarly, the system tells the user which parts can be disposed.

Step 4: Making joints with Power Tools. Similar to slicing operations, the *instruction manual* module renders interactive 3D instructions to assemble and use JigFab's constraints for making all joints. Figure 4 shows the steps for making a finger joint: (a) The user inserts a custom laser-cut template in the router depth transfer box. (b) The plunge router is positioned on top, and the router is pushed down until the router bit touches the depth template and locks the depth control of the router. (c) The user assembles the custom finger joint jig and fixates the work piece inside. (d-e) The user now moves the router within the toolpath to make the finger joint.

Step 5: Assembling the final workpiece. Once the interactive manual is finished, users assemble all parts of the work piece according to the original design (Figure 1d).

Labeling and orienting the work piece

During the slicing operations (step 3), JigFab instructs users to transfer a label, displayed on the 3D rendering, onto the work piece using a pencil. A part with a single label results in two parts after every subsequent cut. Therefore users are instructed to erase the previous label and apply the new one after every cut. Labels consist of a unique name as well as two arrows that specify the orientation of the workpiece. In every operation, JigFab visualizes these labels to instruct the user how to position and orient the work piece.

4 PARAMETRIC JOINTS

In this section, we discuss the use of, and relations between, the 16 joints supported by the JigFab system. Figure 5 shows all types of joints categorized in 4 groups. JigFab's *create joint* module offers explicit support for making various groove, dowel, finger, and mortise & tenon joints, variations of *but joints* and *mitered but joints* are supported indirectly as they are an inherent aspect of designing a work piece, consisting of multiple parts. Each of the four types of joints are parametric and thus customizable through menu options in the JigFab workspace.

By default the groove operation generates a *dado joint* or *rabbit joint* between two selected parts. Applying this operation to a miter cut results in a *mitered rabit joint*. The groove operation supports further fine-tuning using longitudinal and latitudinal offset controls. Adding one latitude offset results in a *barefaced housing joint* or *lap joint* when applied to respectively boards or lumber. To further decorate a groove and make it invisible from the front, a longitudinal offset realizes a *stopped housing joint*.

A special type of groove are tenon and mortise joints as the tenon slides into the mortise from one direction and the mortise entirely embraces the tenon. Therefore, tenon and mortise joints are extremely strong and often used for making frames that have to support significant weight. Using the tenon and mortise operation, JigFab renders such joints

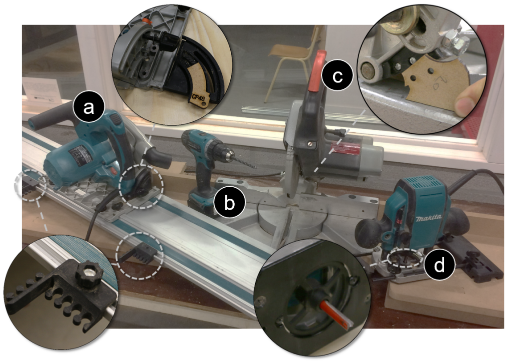

Figure 6: 3D printed adapters to attach computationally designed constraints to (a) the miter saw, (b) the hand-held drill, (c) the compound miter saw, (d) the plunge router.

between two selected parts. A depth offset realizes a *stopped mortise & tenon joint*.

Dowels are popular in carpentry for joining boards because of their easy assembly. JigFab also makes them easy to design. The number of dowels as well as the offsets to the edges of the parts are fine-tuned through menu options. Additionally, we allow for moving dowels from the center of the cross section to the bottom to realize shelf pins.

Finally, *finger* and *dove tail* joints are very strong because of the many contact regions between the parts. However, designing and making these joints traditionally requires true crafting skills. JigFab's finger operation automates the design of intricate finger and dove tail joints between arbitrarily oriented parts.

The joints supported by JigFab have various strengths and decorative qualities depending on the type of material and the design in which they are used. While some joints might be strong enough by themselves, glue can be applied to further reinforce their strength. A complete guide for selecting the optimal joint can be found in Jackson and Day [6] and is beyond the scope if this paper. Future versions of JigFab could internalize this domain knowledge and offer feedback and suggestions on the strength and stability of a work piece similar to the work of Umetani et al. [35].

5 POWER TOOLS AND ADAPTERS

JigFab's approach to computationally fabricated constraints can be applied to many power tools. To support a wide variety of designs, JigFab supports four power tools (Figure 6): a *circular saw*, a *hand-held drill*, a *compound miter saw*, and a *plunge router*. These tools are commonly used in woodworking shops and by maker enthusiasts, and each have their strengths and weaknesses.

To use these power tools in combination with JigFab's constraints, we 3D printed four adapter modules (black parts in Figure 6). (1) Two movable adapters on the ruler guide for attaching fixtures. (2) A copy ring to guide the plunge router along a tool path (also commercially available). (3-4) A pocket on the back of the compound miter saw and the side of the circular saw to configure the bevel angles by inserting a laser-cut template (Figure 3b).

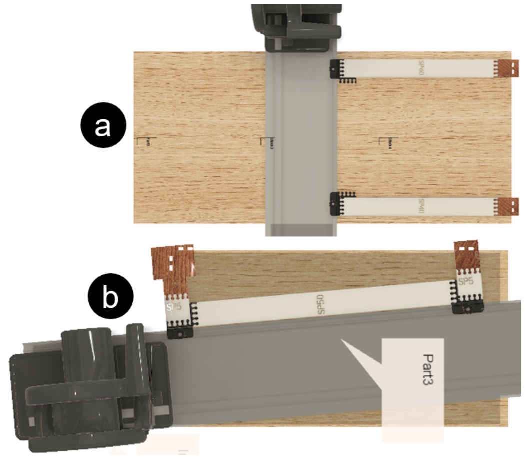

Figure 7: Constraining the circular saw and ruler guide for (a) straight cuts and (b) miter cuts

6 CUSTOM CONSTRAINTS: JIGS, FIXTURES, AND TEMPLATES

In this section, we discuss the strengths and weaknesses of the jigs, fixtures, and templates supported by JigFab. We also discuss how they are used to make joints, and how they are adapted to the specifics of the work piece and design.

Constraining the Circular Saw

Slicing stock material into many parts oftentimes involves long cuts, as shown in step 3 of the walkthrough. For such cuts, JigFab designs constraints for the circular saw and ruler guide. As shown in Figure 7, fixtures attached to the ruler guide precisely position the circular saw with respect to the edges of the work piece. The length of the cut does not require constraints as circular saws are only suitable for making guillotine cuts (cuts that entirely bisect the material). JigFab optimizes for such cuts. As fixtures can get lengthy, JigFab uses predefined elements for the majority of the length. Only the connector element at the edge is custom laser cut to fit the specifics of the work piece design. All fixture elements interconnect tightly with meander shaped endings.

Straight cuts, such as the operation shown in Figure 7a, are parallel to an edge. Miter cuts however, require a fixture in two directions to exactly match the miter angle. JigFab realizes this by designing an end connector to precisely attach the fixture in one of the corners of the work piece as shown in

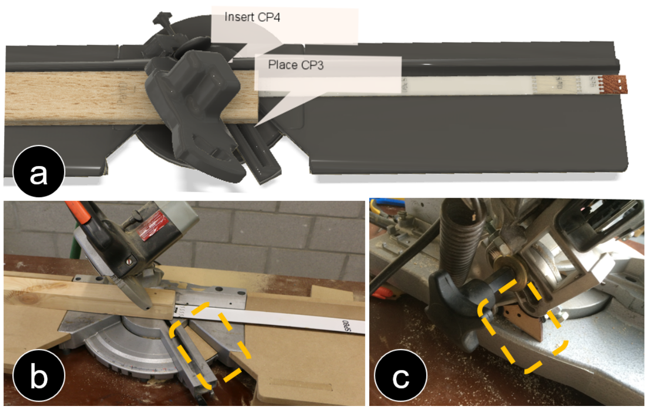

Figure 8: JigFab generates three constraints for the compound miter saw: (a) a fixture for aligning the work piece, and a template for (b) the miter and (c) the bevel angle.

Figure 7b. As the distance between the two fixtures is crucial for the miter cut angle, the two fixtures are spaced using a third fixture attached to the ruler guide. JigFab designs these three fixtures to ensure that only the two end connectors are custom and require laser cutting. The third fixture as well as the other elements of the fixtures are assemblies of predefined elements (Figure 7b).

Straight-bevel and miter-bevel/compound cuts are accomplished by configuring the circular saw's bevel rotational axis with the desired bevel angle. As demonstrated in the walkthrough (Figure 3b), JigFab facilitates configuring the bevel angle of the circular saw using a generated template that fits into an adapter slot of the saw. Once inserted, the user rotates the saw until the template touches the base plate. Figure 6a shows the design of this adapter.

Constraining the Miter Saw

For shorter cuts (up to 20 cm), JigFab designs constraints for the compound miter saw. Figure 8a-b demonstrates how a work piece is configured in longitudinal direction with a fixture from the work piece to the edge of the miter saw's cart. This fixture is similar to the fixture of the ruler guide and mainly consist of predefined elements. The user configures the saw's miter angle by turning the miter axis until the laser-cut template fits. JigFab instructs whether the template needs to fit left or right of the saw. In a similar vein, a generated template fits into the adapter in the back of the miter saw to configure its bevel angle without measuring (Figure 8c).

Constraining the Plunge Router

JigFab suggests using the plunge router for all types of grooves, tenon & mortise joints, as well as finger joints. The router depth is first configured using the router depth transfer box.

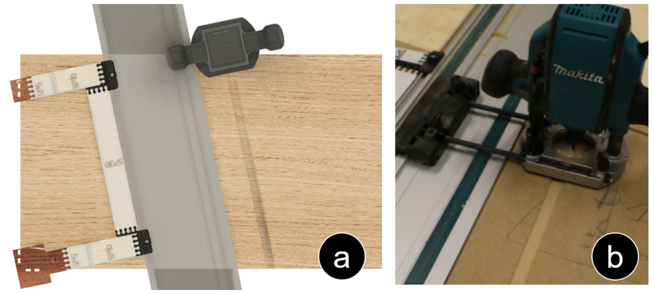

Figure 9: Fabricating a groove using the ruler guide (b) according to the instructions (a).

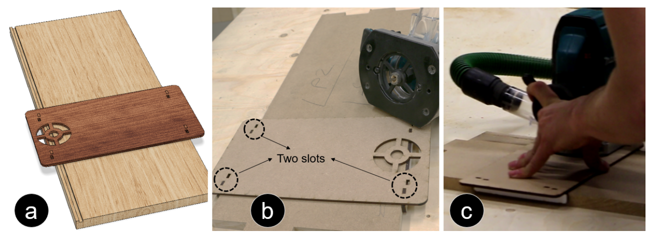

Figure 10: Fabricating a groove using a router edge guide.

As demonstrated in the walkthrough (Figure 4a-b), this involves positioning a generated template in the router depth transfer box and transferring the remaining height to the plunge router.

As shown in Section 5, many variations of grooves exist. All grooves, however, can be considered as a combination of generic notches. Therefore we reformulate the challenge of making various types of grooves to a more universal problem of making notches on different phalanges of the work piece with varying lengths and orientations. JigFab supports three types of dynamic constraints for notches:

a. Router guide constraints use the router in combination with the ruler guide which is especially suitable for long notches. Adapters for attaching the router to the ruler guide are commercially available. JigFab generates fixtures to precisely position and orient the ruler guide using the same strategy as for fixating the circular saw and ruler guide (Figure 9). Furthermore, when the notch design is wider than the milling bit, extra milling iterations are required at offsets. JigFab internalizes this knowledge and adds multiple slots in the end connectors of the fixtures to offset the ruler guide and router.

b. Edge guides slide the router along a parallel edge and are especially suited for making notches in a smaller work piece, such as lumber (Figure 10a-b). In these situations balancing and fixating the ruler guide is challenging. JigFab supports

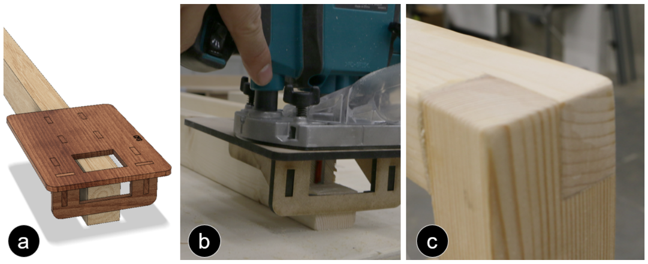

Figure 11: Fabricating grooves with the tool path jig

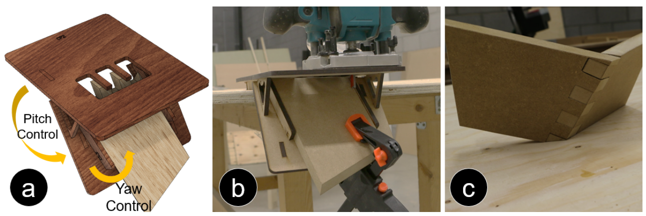

Figure 12: Making highly intricate finger joints using a computationally fabricated jig.

edge guides which, similar to the router guide constraint, supports multiple slots for wide notches (Figure 10c). Edge guide jigs, however, only allow for making notches parallel to an edge of the work piece. Other notch designs require moving the router along more intricate paths. This is the essence of the last type of router jig.

c. Tool path jigs fixture a custom tool path on top of a phalange of the work piece. As they embed a jig and fixture in one constraint, tool path jigs are especially suitable for making notches on small phalanges. Figure 11 shows a tool path jig for making a lap joint. While generating this type of jig, JigFab takes into account the size of the copy ring as well as the placement of dogbone features at inside corners. These overcuts are required as routers can only make round inside corners. Generating such arbitrary tool paths is currently only available in advanced software for large CNC milling machinery. JigFab brings these computational features to power-tools. While this last type of jig can be used for making notches in any configuration, their design is very specific to the notch and work piece which makes them less likely to be reused for making other notch designs.

JigFab uses these three types of constraints to fabricate all supported grooves and mortise & tenon joints (Figure 5). Although it is possible to use a tool path jig for making standard finger and dovetail joints, mitered finger and dovetail joints (Figure 12c), require fixturing the work piece at very specific angles. JigFab therefore supports a parametric jig

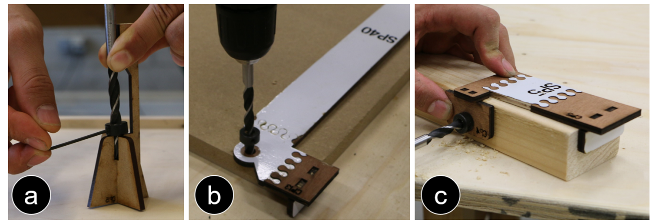

Figure 13: Making a dowel joint requires: (a) transfering the depth of the drill using a template, (b) making a hole in the front and (c) cross face.

specific for making finger and dovetail joints (Figure 12). The parametric finger joint jig integrates a tool path as well as fixtures to mount the work piece at the required orientation. As shown in Figure 12a, joining two parts that have both different pitch and yaw angles result in highly intricate finger joints that would normally require lots of preparation and calculations. Making such decorative joints is considered to be a true crafting skill only feasible for artisans or with the support of multi-axis CNC milling machinery. JigFab makes designing and fabricating such advanced joints available for casual makers using ordinary power tools.

Constraining the hand-held drill

Making dowel joints is most appropriate with a hand-held drill. Depending on the thickness of the material and the length of the dowel, the dowel is positioned deeper in the cross section. Therefore, the drilling depth needs to be configured. JigFab generates a drill depth jig, shown in Figure 13a, to precisely position a drill stop on the drill bit. Furthermore, JigFab designs fixtures to precisely position the drill guide on top of the work piece (Figure 13b). This type of fixture is similar to fixtures for the ruler guide and thus only uses custom laser-cut elements at the edges. For the holes in the cross section, JigFab generates a drill guide with fixtures to the edges of the work piece (Figure 13c).

7 IMPLEMENTATION

We implemented JigFab as a plugin for Autodesk Fusion 360 [5]. As such, all regular modeling features of Fusion are available in addition to the automated design of joints and constraints offered by JigFab. The plugin is written in Python and consists of a sequence of modules and algorithms which are executed in the same order as they are discussed in this section.

Generating Joints

JigFab's *create joint* module automates modeling features within Fusion. The design of every joint starts by computing the interference body between two selected parts. Other

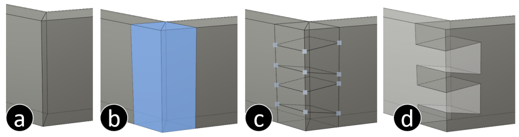

Figure 14: Generating a dove tail joint: (a-b) computing the enclosing volume of all fingers, (c) plane-cuts for subdividing the fingers, (d) the final dove-tail joint

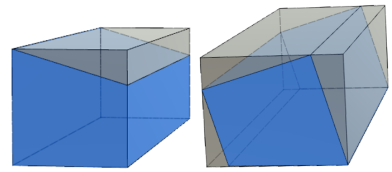

Figure 15: Analyzing the optimal orientation of a work piece by considering all faces

operations are specific to the type of joint (Section 4). The finger/dove tail joint generation first projects the interference body onto the outer faces of both parts and applies a non-linear extrusion (sweep) from one of these projection profiles to the other. This results in an enclosure volume for all fingers (Figure 14b). Plane cuts then subdivide this volume into the desired number of fingers (Figure 14c). Boolean union and difference operations respectively merge or exclude individual fingers from one of the two parts (Figure 14d).

The automated groove and mortise & tenon joint design process starts with a linear extrusion of the interference body. Plane cuts are then applied on the extruded volume according to the longitudinal and latitudinal offsets specified by the user. Finally, the dowel design operation is implemented as a series of cylinder extrusions along the longitudinal direction of the interference body.

Extracting Miter Cuts

In contrast to joints that are explicitly generated by JigFab, slanted shapes, requiring miter cuts, are unknown to JigFab as they result from various modeling operations applied by the user (e.g. sketches with angular edges, chamfer operations, or plane cuts). Therefore the *fabricate* module start with identifying all faces that require miter cuts to go from the part's oriented bounding box to its final shape. During this process, the orientation of every part is optimized to reduce the number of miter cuts, given the available stock material entered by the user. For example, the part in Figure 15 requires one cut in 50x50mm lumber while the same part would require three cuts in 60x50mm lumber when tilted on the slanted face. In this stage, all slanted faces that are part of a joint (e.g. dove tail joints) are discarded as those will be handled while fabricating the joints.

Generating Constraints

For every cut operation and all operations that constitute a joint, JigFab first computes which power tool and type of constraint fits specifics of the joint design and work piece. Many solutions can exist of which our implementation will select the most appropriate. For example, the notch of a rabbit joint (Figure 5) could be fabricated by positioning the plunge router on the side face, cross face, top face, or bottom face. However, some faces are too narrow for the router or would require very long router bits.

Our optimization approach starts with analyzing the geometry of the work piece and extracting all relevant features. For groove joints, features, such as the dimensions of the work piece and edges parallel to the groove are extracted. Together with the features of the groove, such as its length, depth, and width, these properties pass a decision tree, of which the criteria are covered in Section 6, to retrieve the appropriate power tool and type of constraint.

Once the appropriate power tool and type of constraint are known, features for the parametric models are calculated. These features include the geometric features as well as user defined settings, such as the diameter of the router bit and the thickness of the materials, available for laser cutting the custom elements of constraints. The latter parameter is important for constraints that position the power tool on top, and thus offset the tool to the work piece (e.g. the edge guide, tool path jig, and drill guide). For fixtures, JigFab additionally substitute constraints by predefined elements and smaller custom elements.

Automated Panelizing

JigFab panelizes the oriented bounding boxes of all parts on the available items of stock material using the guillotine cutting stock algorithm as described in [7]. This algorithm ensures that rectangular items can be cut using only guillotine cuts. As such, all slicing operations can be done with a circular saw and miter saw.

All slicing operations are then ordered to minimize the number of cuts by prioritizing cuts with the most collinear edges. JigFab compensates for the thickness of the saw by adding half of the saw thickness to the edges of all parts. The stock material is offset with half of the saw thickness to compensate for the edges of the parts located at the boundary of the stock material. For every slicing operation, JigFab generates constraints according to the process described in the previous section.

Exporting Models

In the final step of the *fabricate* module, JigFab exports all custom elements of constraints to DXF files for laser cutting

Material for work piece:	1x 122x244x18mm plywood 2x 300x62x38mm CLS/SLS	2x 122x244x18mm MDF 1x 122x244x6mm MDF
Material for constraints:	1x 60x30x4mm MDF	3x 60x30x4mm MDF

Figure 16: Several constructions fabricated with JigFab, including a table and a display cabinet with drawer.

and STL files for visualization in the instruction manual. This process uses Fusion's parametric modeling environment to pass all computed parameters to our parametric models. Finally, labels are embossed in all parts of the work piece and exported for rendering in the instruction manual.

Starting the *instruction manual* module triggers a step-by-step interactive manual that first guides the user through the slicing and miter operations, and then shows how to perform the joinery operations. Note that the 3D models used in the visualization of the joinery operations always visualize all joints of the part, as the STL models are exported at the very end. Joints that are fabricated in subsequent operations could be removed from the visualization by exporting intermediate STL files while designing joints. Boolean operations between the final and intermediate STL models are then required to render only the desired joints while preserving the size of the final STL model. Directly rendering intermediate STL models, during the joint design process, is not possible as these parts are still subject to extrusions.

8 EXAMPLE CONSTRUCTIONS

To validate our approach, we fabricated several advanced constructions with JigFab, including a table and a display cabinet with drawer shown in Figure 16. Fabricating such constructions without JigFab would require large-scale industrial CNC machinery or solid woodworking skills. In contrast, our example constructions, were solely made with JigFab's supported tools and constraints. The person involved in the fabrication process had very limited practical woodworking experience. After every fabrication session, which lasted a maximum of 3 hours, we measured the parts before assembly. All parts approximated the digital version within 2 millimeters. This error is acceptable in wood working and similar to the accuracy of 1/16 inch (1.58mm), desired by many carpenters. Figure 16 also reports on the materials used for the work piece and constraints. Although JigFab supports reusing custom laser-cut constraints across projects, we did not use this feature to report on JigFab's material consumption per project. We refer the reader to the supplementary material, attached to this submission, for renderings of the entire instruction manual for making the cabinet.

9 DISCUSSION AND FUTURE WORK

While the current version of JigFab is tailored for casual makers, without significant expertise with power tools, our system might also appeal to crafting experts as it designs highly complex joints and corresponding jigs that otherwise require many calculations, modeling, and crafting skills (Figure 12). In the future, we plan to conduct workshops with casual and experienced makers to gather in-depth feedback on JigFab's features and workflow.

As our approach uses commercially available power tools, small custom 3D printed adapters attached to these power tools make them compatible with JigFab's constraints. When using a different brand or version of power tool its dimension and angular constraints have to be configured in a settings file. Additionally, the adapters have to be redesigned once, to make the power tool compatible with all types of constraints. In the future, these adapters can be shared among makers through online platforms, such as Thingiverse. Alternatively, when computationally fabricated constraints become more widely used, manufacturers of power tools could embed one standardized connector.

Although JigFab already supports common power tools and many joints and constraints, more can be added in the future. For example, we require all boards and lumber, entered as material stock for slicing, to be perfectly square from the start. Although this service is offered by many DIY shops and timber trades, future versions of JigFab could support features to facilitate using a try square to rectify the work piece at the start. Additional power tools with appropriate constraints, such as a bandsaw or jigsaw could allow for making more complex concave shapes and curves. To support such new types of power tools, one needs to design new types of parametric constraints and model their characteristics in JigFab's decision tree (Section 6). Finally, JigFab could also support makers in the assembly of their design [17] or provide suggestions to improve its stability [35, 38].

10 CONCLUSION

As personal fabrication continues to rise in popularity and highly personalized artifacts become commonplace, we expect people to also desire this for larger-scale constructions. JigFab takes a new approach to large-scale fabrication and automates the design of constraints to facilitate building constructions with common power tools. Starting from a digital version of the construction, our system generates jigs, fixtures, and templates that constrain and configure the work piece and power tool during every operation. Our constraints are custom to the work piece and therefore easy to handle according to our interactive step-by-step manual. With this approach, JigFab empowers casual makers to move from raw stock material to constructions embedding intricate woodworking joints. Building such structures is normally reserved for artisans or automated with advanced CNC machinery.

REFERENCES

[1] Harshit Agrawal, Udayan Umapathi, Robert Kovacs, Johannes Frohnhofen, Hsiang-Ting Chen, Stefanie Mueller, and Patrick Baudisch. 2015. Protopiper: Physically Sketching Room-Sized Objects at Actual Scale. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. ACM, New York, NY, USA, 427–436. https://doi.org/10.1145/2807442.2807505

[2] Boxes.py. 2018. Create boxes and more with a laser cutter! https://www.festi.info/boxes.py. Accessed: 2018-09-20.

[3] Xiang 'Anthony' Chen, Jeeeun Kim, Jennifer Mankoff, Tovi Grossman, Stelian Coros, and Scott E. Hudson. 2016. Reprise: A Design Tool for Specifying, Generating, and Customizing 3D Printable Adaptations on Everyday Objects. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16)*. ACM, New York, NY, USA, 29–39. https://doi.org/10.1145/2984511.2984512

[4] Chi-Wing Fu, Peng Song, Xiaoqi Yan, Lee Wei Yang, Pradeep Kumar Jayaraman, and Daniel Cohen-Or. 2015. Computational Interlocking Furniture Assembly. *ACM Trans. Graph.* 34, 4, Article 91 (July 2015), 11 pages. https://doi.org/10.1145/2766892

[5] Autodesk Inc. 2018. Fusion 360 CAD/CAM software connects your entire product design development process in a single tool. https://www.autodesk.com/products/fusion-360/overview. Accessed: 2018-09-20.

[6] Albert Jackson, David Day, Ian Kearey, and Robin Harris. 1995. *Good wood joints*. HarperCollins.

[7] Jukka Jylänki. 2010. A thousand ways to pack the bin-a practical approach to two-dimensional rectangle bin packing. *retrived from http://clb. demon. fi/files/RectangleBinPack. pdf* (2010).

[8] Jeeeun Kim, Anhong Guo, Tom Yeh, Scott E. Hudson, and Jennifer Mankoff. 2017. Understanding Uncertainty in Measurement and Accommodating Its Impact in 3D Modeling and Printing. In *Proceedings of the 2017 Conference on Designing Interactive Systems (DIS '17)*. ACM, New York, NY, USA, 1067–1078. https://doi.org/10.1145/3064663.3064690

[9] Jarrod Knibbe, Tovi Grossman, and George Fitzmaurice. 2015. Smart Makerspace: An Immersive Instructional Space for Physical Tasks. In *Proceedings of the 2015 International Conference on Interactive Tabletops & Surfaces (ITS '15)*. ACM, New York, NY, USA, 83–92. https://doi.org/10.1145/2817721.2817741

[10] Robert Kovacs, Anna Seufert, Ludwig Wall, Hsiang-Ting Chen, Florian Meinel, Willi Müller, Sijing You, Maximilian Brehm, Jonathan Striebel, Yannis Kommana, Alexander Popiak, Thomas Bläsius, and Patrick Baudisch. 2017. TrussFab: Fabricating Sturdy Large-Scale Structures on Desktop 3D Printers. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 2606–2616. https://doi.org/10.1145/3025453.3026016

[11] Yuki Koyama, Shinjiro Sueda, Emma Steinhardt, Takeo Igarashi, Ariel Shamir, and Wojciech Matusik. 2015. AutoConnect: Computational Design of 3D-printable Connectors. *ACM Trans. Graph.* 34, 6, Article 231 (Oct. 2015), 11 pages. https://doi.org/10.1145/2816795.2818060

[12] Benjamin Lafreniere, Tovi Grossman, Fraser Anderson, Justin Matejka, Heather Kerrick, Danil Nagy, Lauren Vasey, Evan Atherton, Nicholas Beirne, Marcelo H. Coelho, Nicholas Cote, Steven Li, Andy Nogueira, Long Nguyen, Tobias Schwinn, James Stoddart, David Thomasson, Ray Wang, Thomas White, David Benjamin, Maurice Conti, Achim Menges, and George Fitzmaurice. 2016. Crowdsourced Fabrication. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16)*. ACM, New York, NY, USA, 15–28. https://doi.org/10.1145/2984511.2984553

[13] Manfred Lau, Akira Ohgawara, Jun Mitani, and Takeo Igarashi. 2011. Converting 3D Furniture Models to Fabricatable Parts and Connectors. In *ACM SIGGRAPH 2011 Papers (SIGGRAPH '11)*. ACM, New York, NY, USA, Article 85, 6 pages. https://doi.org/10.1145/1964921.1964980

[14] Manfred Lau, Akira Ohgawara, Jun Mitani, and Takeo Igarashi. 2011. Converting 3D Furniture Models to Fabricatable Parts and Connectors. In *ACM SIGGRAPH 2011 Papers (SIGGRAPH '11)*. ACM, New York, NY, USA, Article 85, 6 pages. https://doi.org/10.1145/1964921.1964980

[15] Manfred Lau, Greg Saul, Jun Mitani, and Takeo Igarashi. 2010. Modeling-in-context: User Design of Complementary Objects with a Single Photo. In *Proceedings of the Seventh Sketch-Based Interfaces and Modeling Symposium (SBIM '10)*. Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 17–24. http://dl.acm.org/citation.cfm?id=1923363.1923367

[16] Danny Leen, Raf Ramakers, and Kris Luyten. 2017. StrutModeling: A Low-Fidelity Construction Kit to Iteratively Model, Test, and Adapt 3D Objects. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST '17)*. ACM, New York, NY, USA, 471–479. https://doi.org/10.1145/3126594.3126643

[17] Wilmot Li, Maneesh Agrawala, Brian Curless, and David Salesin. 2008. Automated Generation of Interactive 3D Exploded View Diagrams. In *ACM SIGGRAPH 2008 Papers (SIGGRAPH '08)*. ACM, New York, NY, USA, Article 101, 7 pages. https://doi.org/10.1145/1399504.1360700

[18] Jeffrey I Lipton, Adriana Schulz, Andrew Spielberg, Luis Trueba, Wojciech Matusik, and Daniela Rus. [n. d.]. Robot Assisted Carpentry for Mass Customization. ([n. d.]).

[19] Shiran Magrisso, Moran Mizrahi, and Amit Zoran. 2018. Digital Joinery For Hybrid Carpentry. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 167, 11 pages. https://doi.org/10.1145/3173574.3173741

[20] MakerCase. 2018. Easy Laser Cut Case Design. http://www.makercase.com/. Accessed: 2018-09-20.

[21] Stefanie Mueller, Tobias Mohr, Kerstin Guenther, Johannes Frohnhofen, and Patrick Baudisch. 2014. faBrickation: Fast 3D Printing of Functional Objects by Integrating Construction Kit Building Blocks. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems (CHI EA '14)*. ACM, New York, NY, USA, 187–188. https://doi.org/10.1145/2559206.2582209

[22] Yeonjoo Oh, Gaberial Johnson, Mark Gross, and Ellen Yi-Luen Do. 2006. The designosaur and the furniture factory. In *Design Computing and CognitionâĂŹ06*. Springer, 123–140.

[23] Hutchison Paul. 2018. Inkscape tabbed box generator plugin. https://github.com/paulh-rnd/TabbedBoxMaker. Accessed: 2018-09-20.

[24] Huaishu Peng, Rundong Wu, Steve Marschner, and François Guimbretière. 2016. On-The-Fly Print: Incremental Printing While Modelling. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 887–896. https://doi.org/10.1145/2858036.2858106

[25] Huaishu Peng, Amit Zoran, and François V. Guimbretière. 2015. D-Coil: A Hands-on Approach to Digital 3D Models Design. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 1807–1815. https://doi.org/10.1145/2702123.2702381

[26] Alec Rivers, Ilan E. Moyer, and Frédo Durand. 2012. Position-correcting Tools for 2D Digital Fabrication. *ACM Trans. Graph.* 31, 4, Article 88, 7 pages. https://doi.org/10.1145/2185520.2185584

[27] Greg Saul, Manfred Lau, Jun Mitani, and Takeo Igarashi. 2011. SketchChair: An All-in-one Chair Design System for End Users. In *Proceedings of the Fifth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '11)*. ACM, New York, NY, USA, 73–80. https://doi.org/10.1145/1935701.1935717

[28] Eldon Schoop, Michelle Nguyen, Daniel Lim, Valkyrie Savage, Sean Follmer, and Björn Hartmann. 2016. Drill Sergeant: Supporting Physical Construction Projects Through an Ecosystem of Augmented Tools. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '16)*. ACM, New York, NY, USA, 1607–1614. https://doi.org/10.1145/2851581.2892429

[29] SketchList3D. 2018. Cabinet Design Software for Custom Woodworking Professionals. https://sketchlist.com. Accessed: 2018-09-20.

[30] Peng Song, Bailin Deng, Ziqi Wang, Zhichao Dong, Wei Li, Chi-Wing Fu, and Ligang Liu. 2016. CofiFab: Coarse-to-fine Fabrication of Large 3D Objects. *ACM Trans. Graph.* 35, 4, Article 45 (July 2016), 11 pages. https://doi.org/10.1145/2897824.2925876

[31] Peng Song, Chi-Wing Fu, Yueming Jin, Hongfei Xu, Ligang Liu, Pheng-Ann Heng, and Daniel Cohen-Or. 2017. Reconfigurable Interlocking Furniture. *ACM Trans. Graph.* 36, 6, Article 174 (Nov. 2017), 14 pages. https://doi.org/10.1145/3130800.3130803

[32] Rundong Tian, Sarah Sterman, Ethan Chiou, Jeremy Warner, and Eric Paulos. 2018. MatchSticks: Woodworking Through Improvisational Digital Fabrication. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 149, 12 pages. https://doi.org/10.1145/3173574.3173723

[33] Handibot: Smart Power Tool. 2018. https://handibot.com. Accessed: 2018-09-20.

[34] Cesar Torres, Wilmot Li, and Eric Paulos. 2016. ProxyPrint: Supporting Crafting Practice Through Physical Computational Proxies. In *Proceedings of the 2016 ACM Conference on Designing Interactive Systems (DIS '16)*. ACM, New York, NY, USA, 158–169. https://doi.org/10.1145/2901790.2901828

[35] Nobuyuki Umetani, Takeo Igarashi, and Niloy J. Mitra. 2012. Guided Exploration of Physically Valid Shapes for Furniture Design. *ACM Trans. Graph.* 31, 4, Article 86 (July 2012), 11 pages. https://doi.org/10.1145/2185520.2185582

[36] Christian Weichel, Manfred Lau, David Kim, Nicolas Villar, and Hans W. Gellersen. 2014. MixFab: A Mixed-reality Environment for Personal Fabrication. In *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 3855–3864. https://doi.org/10.1145/2556288.2557090

[37] Yong-Liang Yang, Jun Wang, and Niloy J Mitra. 2014. Mesh2Fab: Reforming Shapes for Material-specific Fabrication. *arXiv preprint arXiv:1411.3632* (2014).

[38] Jiaxian Yao, Danny M. Kaufman, Yotam Gingold, and Maneesh Agrawala. 2017. Interactive Design and Stability Analysis of Decorative Joinery for Furniture. *ACM Trans. Graph.* 36, 2, Article 157a (March 2017). https://doi.org/10.1145/3054740

[39] Clement Zheng, Ellen Yi-Luen Do, and Jim Budd. 2017. Joinery: Parametric Joint Generation for Laser Cut Assemblies. In *Proceedings of the 2017 ACM SIGCHI Conference on Creativity and Cognition (C&C '17)*. ACM, New York, NY, USA, 63–74. https://doi.org/10.1145/3059454.3059459

[40] Amit Zoran, Nan-Wei Gong, Roy Shilkrot, Shuo Yan, and Pattie Maes. 2015. Cutting Edge Vision: Metal Embedded Optics for Smart Knives. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '15)*. ACM, New York, NY, USA, 185–185. https://doi.org/10.1145/2702613.2732495

[41] Amit Zoran and Joseph A. Paradiso. 2013. FreeD: A Freehand Digital Sculpting Tool. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, New York, NY, USA, 2613–2616. https://doi.org/10.1145/2470654.2481361