Drill Sergeant: Supporting Physical Construction Projects through an Ecosystem of Augmented Tools

Michelle Nguyen
Eldon Schoop
Mitchell Karchemsky
Valkyrie Savage
Björn Hartmann
Sean Follmer

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2016-90
http://www.eecs.berkeley.edu/Pubs/TechRpts/2016/EECS-2016-90.html

May 13, 2016

Acknowledgement

I would like to thank Professor Bjoern Hartmann for his guidance during my time as an undergraduate, and now, as a graduate student. I would also like to thank the Drill Sergeant team for all of their hard work and friendship throughout the project. Finally, I would like to thank my family for their continuous love and support.

This work was supported in part by TerraSwarm, one of six centers of STARnet, a Semiconductor Research Corporation program sponsored by MARCO and DARPA. This material is also based on work supported by the NSF under Grant No. DGE 1106400.

Drill Sergeant: Supporting Physical Construction Projects through an Ecosystem of Augmented Tools

Michelle Nguyen[†], **Eldon Schoop**[†], **Mitchell Karchemsky**[‡], **Valkyrie Savage**[†], **Bjoern Hartmann**[†] **Sean Follmer**[§]
†: UC Berkeley EECS, ‡: UC Berkeley Cognitive Science, §: Stanford Mechanical Engineering
{aimichelle,eschoop,mkarch}@berkeley.edu,
{valkyrie,bjoern}@eecs.berkeley.edu, sfollmer@stanford.edu

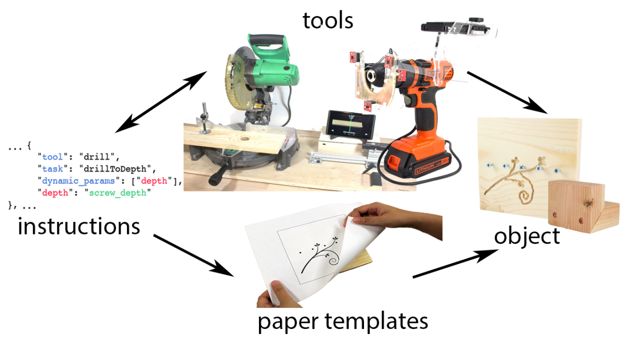

Figure 1. In Drill Sergeant, tools and model specifications inform and define each other as a user builds construction macros with the aid of the tools and model-generated paper templates.

ABSTRACT

Woodworking tutorials are commonly used to gain proficiency with power tools. However, these tutorials' goals generally aim to assist users with constructing an end product, as opposed to developing their confidence and technique with tools. We present an ecosystem of smart tools which helps users gain confidence, develop technique, and construct woodworking projects by providing guidance and feedback through skill-building activities and macros of common building tasks. We demonstrate how such techniques are enabled by augmenting common workshop tools (drill/driver, saw) with measurement, state sensing, and visual feedback, and describe the design space of such augmentations. An evaluation with 17 novices shows greater improvement of some tacit skills when using augmented tools compared to unaugmented tools. We also validate the utility and flexibility of a smart tool ecosystem through reflections on a series of author-created design examples.

ACM Classification Keywords

H.5.2. Information Interfaces and Presentation (e.g. HCI): Smart Tools

Author Keywords

Smart Tools, Multi-Machine Ecosystems, Learning, Instructional Systems, DIY

INTRODUCTION

Builders—home improvement enthusiasts, woodworkers, and DIYers—craft physical artifacts using standard workshop tools. A common way for novices to acquire expertise with tools is through building projects based on tutorials. Project tutorials are published in books, e.g. [28, 38], and online, as in Instructables[1]. However, these tutorials present several major shortfalls for novices. They consist of *static* instructions, leaving novices with a non-trivial translation from written steps to corresponding actions [45]. Furthermore, although project tutorials are generally composed of common construction tasks, such as cutting boards to length or fastening parts with screws, it may be challenging to apply these operations for use in different contexts, such as personalized projects and home repairs. Physical tutorials also share several flaws with software tutorials [17]: as the volume of tutorials grow, users may find it intimidating and difficult to identify a suitable entry point into the craft, as production quality, tool requirements, and end results vary [18].

Tacit knowledge is described by Polyani as knowledge which is difficult to formalize and communicate, such as riding a bicycle [32]. Although tacit knowledge is required to successfully execute machining operations, many tutorials assume prior knowledge and instead focus on assisting users with building an end product. Similar to using training wheels, builders may employ various tricks throughout the construction process to compensate for tacit knowledge, such as using the reflective surface of a Compact Disc (CD) to better visualize a drill's orientation[2]. As with the reported shortfalls of training wheels, these tricks may limit the tools' machining capacity or, due to reliance on an external aid, may even hinder development of the necessary intuition.

Prior work has investigated building *individual* smart tools to assist with a single operation [34, 51], and multi-tool use

[1]http://www.instructables.com/

[2]http://www.popularmechanics.com/home/how-to-plans/how-to/g1534/3-tips-for-drilling-super-straight-holes/

through sensing [2, 24]. Other works have explored delivering augmented reality interfaces to *overlay* instructional information onto real-world scenes [13]. We aggregate and extend these works, mapping dynamic software tutorial concepts [6, 10, 11] onto training activities and construction tasks with an ecosystem of augmented traditional tools that work together, inform each others' use, and provide visual guiding information and real-time feedback on progress and mistakes (see Figure 1).

We present *Drill Sergeant*, an ecosystem of augmented power tools which support users in developing confidence and technique through *skill-building activities*; and a library of *assembly macros*, which describe parametrized tasks with our tools and enable novice users to apply their new domain knowledge to personal projects.

We augment commercially-available tools with sensors, displays, and wireless connectivity. The sensors enable real-time tracking of a tool's state to gauge a user's progress or technique while executing an operation. Tool interfaces provide situated guidance by visually expressing implicit tool properties, such as drill orientation or steadiness of a saw stroke, and comparisons of actual tool state and desired outcomes. Our current collection of smart tools includes a pose-sensing drill with projected, spatial augmented reality feedback, a compound miter saw with tablet-based feedback, and an off-the-shelf digital distance measurement tool. (See Figure 7).

Drill Sergeant facilitates mastery over individual tool operations (e.g., drilling at specific angles, making steady cuts) through skill-building activities. During these activities, users work through repetitive exercises, during which they are provided with both real-time and after-the-fact feedback to help improve technique. As users progress, feedback is gradually reduced to cement tacit learning.

The system's assembly macros describe instructions and parametrized machining operations for tasks commonly performed in construction projects (e.g., cutting a board to length, drilling two holes at a relative distance, creating a scarf joint). Macros can incorporate information from a user's environment and preceding actions into a digital assembly model. Templates are generated from the model and may be printed to assist with alignment and tool positioning for tasks within the macro. Macros may be used atomically, to aid a user in a free-form project, or be chained together to create complete, parametrized assemblies.

In summary, this paper offers three core contributions:

- a set of augmented power tools that visually report state information, and the design space of augmentation strategies such tools
- a system for skill-building activities to improve users' tool handling by providing real-time feedback, validated by a user evaluation with novices
- a novel architecture for a smart tool ecosystem connected to parametric design files that generate paper templates and specify how to perform assembly macros, demonstrated through example objects

RELATED WORK

Drill Sergeant builds on prior work in three primary areas: tutorials, smart tools, and modeling.

Tutorials and Instructions

Physical Task Tutorials

Studies of how people use existing published tutorials for physical DIY projects suggest a significant gap between static instruction and actual practice [45, 46], especially when instructions are transferred to settings not identical to the ones described in a tutorial.

One approach augments the workspace to deliver instructions *in situ*. This is popular in kitchen scenarios—e.g., CounterActive [15], Mimicook [37] and French Kitchen [14] which display instructions and track progress through recipes using computer vision and accelerometer-augmented utensils. Drill Sergeant does not constrain users to a fixed-size countertop workspace: the tools are portable and self-contained, and they deliver real-time feedback on the performance of a particular step, enabling *digital apprenticeship* [5].

Other systems, e.g., to aid in furniture [2] or block assemblies [12], also track progress through multi-step instructions. These systems use a *known* set of building blocks or parts—in contrast, Drill Sergeant guides the user through subtractively fabricating parts from raw materials. Beyond recognizing which tool someone is using [24], we provide feedback that compares desired outcomes for each step to a user's actual performance.

Perhaps closest to our domain is Smart Makerspace, which adds sensing to tools placed on a tabletop display [17]. Drill Sergeant explores realtime *in situ* feedback, allowing for use in any environment, coupled with integrated high precision sensing of tools. We also focus on guiding users through assembly macros which can be customized and informed by the user's context, as opposed to static tutorials.

Software Tutorials

We are inspired by prior work on *software* tutorials. Since thoroughly instrumenting software is often feasible, user actions can be recorded precisely. Such action traces can be used to generate and track progress through tutorials [3, 6, 10, 11, 16]. Some research techniques re-target a given tutorial—e.g., adapting red-eye removal to novel input images [4]. User-defined macros which capture the direct manipulation of software tools have also been used in tutorial contexts [20]. Similarly, Drill Sergeant records users' physical interactions with tools to guide and track progress while building assembly macros.

Skill-Building Activities

Educational game design guidelines cover the basic steps for creating effective learning games: measuring behavior, analyzing behavior changes, and providing feedback [23]. Skill-building through repetition and corrective feedback is commonly explored in physical domains such as athletic training [25, 19, 31] and motion rehabilitation [43]. Repeated practice and relevant feedback on progress or mistakes can also help

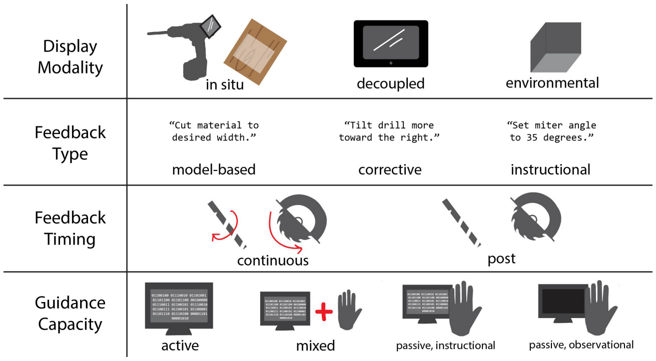

Figure 2. The design space for tool augmentations. Display modality, feedback type and timing, and guidance capacity should be varied according to the tool and its use.

with software applications [7]. We apply these approaches to improving tool technique and skills.

Augmented Reality (AR) for Instruction
One of AR's main benefits is spatially linking information and process: this can aid an inexperienced operator in detecting problems with running machines [30], or allow experts to aid remote novices in assembly tasks [29]. AR can augment information on blueprints [42], or repurpose and project information from plans in heads-up displays [26]. We leverage tool-mounted AR to guide user tool handling, both as training and to aid completion of desired macros.

Smart tools
Zoran, et al., describe the applications of smart tools , outlining their many uses in technical disciplines such as medicine or fabrication, focusing on *individual* tools and their link with digital geometry [52]. A handheld tool can *correct* a user's toolpath to follow the model closely [34], or *guide* the user to sculpt a model from a block of material [33, 53]. Protopiper supports freehanding and later digitizing large structures, and is able to store particular dimensions to ensure, for example, that a box has equal-length sides [1]. Research has also explored connected measurement devices that exchange measured data with CAD software [22, 47]. Smart tools with some of these capabilities already appear commercially [36, 40], indicating the potential of augmented tools for the everyday builder. In contrast to prior work, Drill Sergeant permits a full *ecosystem* of tools to work in concert to process raw parts from multiple types of materials, enabling more complex projects than a single measurement or cutting tool working alone.

CAD Modeling in Physical Contexts
We take inspiration from prior work exploring CAD modeling *in context*—defining and modifying models through actions in physical space rather than manipulation on a screen. This context can come from bringing physical objects into an AR modeling booth [48], or from recording a user's interactions with the tool [49]. For 2D design, tangibles can be manipulated and tracked on a touch screen [39], or imaged *in situ* on the bed of a laser cutter [9, 27]. Drill Sergeant object dimensions come from users physically cutting boards or using digital measurement tape.

THE DESIGN SPACE OF AUGMENTED TOOLS
Our review or prior art indicates the benefit of using augmented tools to help users develop tacit fabrication skills, quickly fabricate design prototypes, and incorporate contextual information from digital models. However, no general guidelines exist for designing augmentations for tools. In order to guide the creation of smart tools, we classify the *display modalities*, *types of feedback*, and *guidance capacity* these tools can provide. (See Figure 2).

Display Modalities
Fishkin describes embodiment as how closely a system's input is tied to its output [8]. Displays for augmented tools can provide *in situ*, *decoupled*, or *environmental* output, according to its degree of embodiment. *In situ* displays incorporate forms of *nearby* and *full* embodiment, showing information on the work material [33] or on the tool itself [34], respectively. These methods maintain a user's focus on the building operation and UI simultaneously. *Decoupled* displays, a form of *distant* embodiment, can be carried or worn by the user, e.g., [21], or they can be part of an augmented environment, e.g., via tabletop displays [17] or the ambient workspace [37]. *Environmental* embodiment provides information through the user's surroundings, such as audio [14].

We chose both *in situ* and *decoupled* modalities, informed by the tools' handling and safety implications. Handheld tools similar to our drill can be used in many different orientations and locations. We therefore use projected feedback in the same locality as the cutting bit, enabling the system to display information in a user's field of view while drilling. In contrast, as it is relatively stationary, our miter saw uses a *decoupled* display, a tablet positioned near the area where a user clamps wood for cutting. This strategy intentionally directs attention to the tablet during setup, then disables the display while cutting to prevent safety risks associated with operating power saws without complete focus.

Feedback
Feedback and information presented by augmented tools varies based on application, comprising *model-based*, *corrective*, and *instructional* feedback. *Model-based* feedback links tool actions to a digital model, whether updating it [48], or using the digital model to specify tooling operations [51]. *Corrective* feedback provides a user with deeper insights into a tool's intrinsic status, which can be used to improve a user's skill [5] or inform a user of otherwise unintuitive information [30]. *Instructional* feedback consists of tutorial content which is delivered through tools, as in through the smart drill's projector, or via the environment [17].

Each type of feedback can be delivered *after* a machining task, or delivered *continuously* while machining. *Continuous* feedback may facilitate a closed feedback loop between user and tool by comparing an expected outcome with current state, e.g. comparing our drill's orientation to a perpendicular reference point. *Post-task* feedback can be used to reinforce learning from mistakes during assembly tasks [5].

Augmented tools in Drill Sergeant employ all three types of feedback during skill-building activities and construction

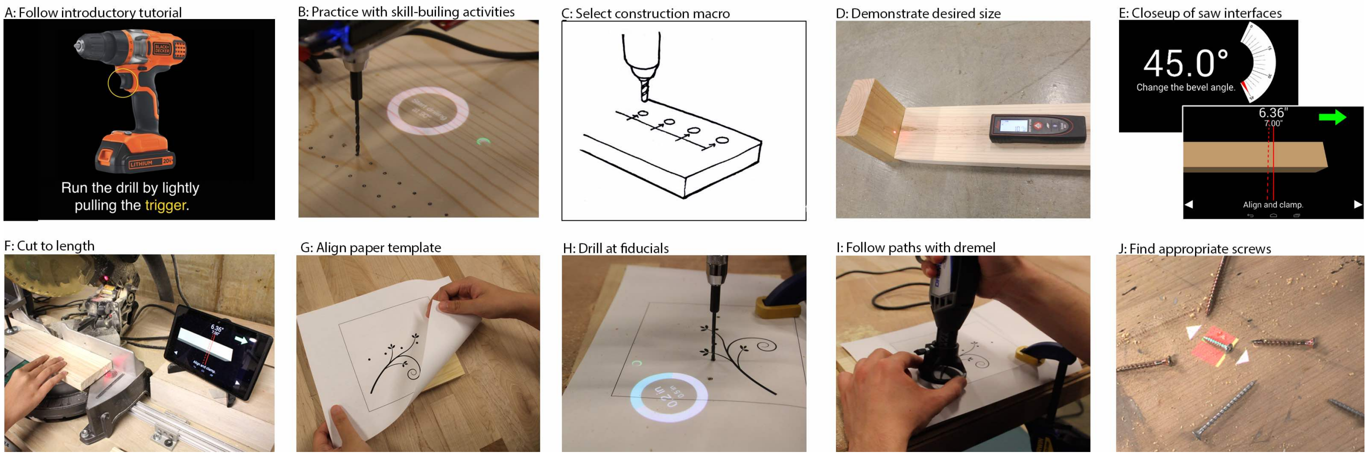

Figure 3. To construct a keyrack, a user follows the drill's introductory tutorial (a), practices drilling with skill-building activities (b), selects a macro (c) and sizes it, measuring their desired final location (d). Next, the user cuts boards to length (e,f), aligns the paper template (g), locates drill positions (h), engraves a decorative pattern (i), and identifies the correct fasteners (j).

macros. . Tools display tutorial information as *instructional* feedback. *Model-based* and *corrective* feedback are outputted *continuously* in the drill's interface to correct users' technique while machining to model specifications. As another example, our miter saw's tablet interface provides *post-task*, *corrective* feedback during its steadiness skill-building activity, providing after-the-fact information to help users improve their steadiness.

Guidance Capacity

Guidance capacity refers to the ability a tool's augmentations has to influence its cutting or modeling path. Tools with *active* guidance maintain full control over the position of their cutting or modeling tool, e.g., standard CNC mills and lathes, and interfaces which augment closed, numerically controlled systems [27, 30]. On the other end of the spectrum, tools with *passive* guidance are controlled completely by a user. Passively guided tools can be *observational*, providing only a platform to sense state and gauge technique [5], or *instructional*, using state sensing to explicitly output tutorial information [17]. Tools with *mixed* guidance provide a mixed-initiative approach, enabling users to deform a parametric model to suit their taste or design goals [51].

Our smart drill and saw utilize *passive, instructional* guidance, relying on the user to actuate all tooling tasks. This approach is in line with the intent to help users build tacit knowledge, as opposed to creating a perfect artifact using *mixed* guidance [34].

USING DRILL SERGEANT

We introduce a running scenario to demonstrate the different interactions enabled by Drill Sergeant's tools.

A novice builder wishes to construct a personalized key rack out of common "1 x 6" lumber [3]. However, they have no experience with using a handheld power drill.

Building Confidence: The builder loads up the drill's introductory tutorial. This tutorial guides them through the anatomy and use of the drill, asking them to perform basic operations (e.g., pulling the trigger, increasing the torque) (Figure 3A).

Skill-Building Activities: Once the builder feels comfortable with basic drill operation, they select skill-building activities to assist them with executing macros required to build the key rack. For example, the activity for perpendicular drilling instructs the builder to bore holes into some scrap material, while providing real-time feedback of the drill's orientation (Figure 3B). As the builder begins to develop an intuition for drilling perpendicular holes, the frequency of the feedback recedes. The builder practices until they feel confident about making their key rack.

Adapting to User Preferences: To make the key rack panel, the builder chooses a construction macro to help them cut their wood to a specific length. The desired width of the key rack is customized using a digital measuring tape on the wall where they wish to hang the key rack. As indicated on the tablet UI, they insert a board into the compound miter saw. The saw helps them align the board to cut to the correct length, displaying a live correction interface on their tablet (Figure 3F).

Cross-Tool Operations: The builder wants 5 evenly-spaced pilot holes where they will drive the screws used to hang their belongings. As decoration, they also want a floral design engraved on the front of the key rack. To do so, they utilize the "evenly-spaced holes" and "center vector" macros, supplying the corresponding macros with their desired number of pilot holes and vector drawing. Once instructed, the builder retrieves the generated paper template from the printer, aligns the template to the panel, and adheres using spray glue (Figure 3G). The template contain 5 evenly-spaced fiducial markings, indicating drill points, and the correctly-positioned and scaled vector drawing which the builder can follow with a dremel (Figure 3H,I)—the paper template thus facilitates subsequent operations with other tools.

[3] The actual dimensions of these boards are 0.75"x5.5"/19x140mm

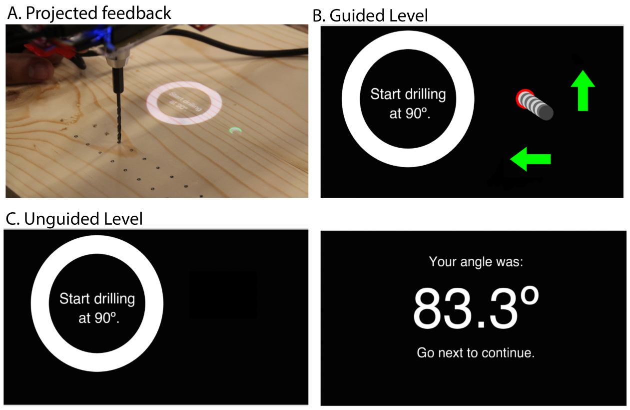

Figure 4. In the drill's skill-building activity for drilling at specific angles, feedback is displayed via the tool's display (a) in real-time (b) or after-the-fact (c).

Finding Parts: After engraving and drilling, the builder selects a macro to help them identify several correctly-sized screws to hold their keys—the smart drill projects a perspective- and size-corrected image of the desired screw size, giving the builder a visual template to identify correct parts (Figure 3J). With the assistance of the drill's depth feedback, they can drive the screws at precise, even depths. The key rack completed, our builder proudly hangs it by their front door.

IMPLEMENTATION

To provide feedback and guidance for training activities and macros, tools are outfitted with sensors, displays and wireless communication. Sensor data represent a tool's internal state, indicating when operations are being executed and whether they are executed correctly (*correct feedback*). Sensors may also collect data for measurements relating to desired material sizes and assembly characteristics (*model-based feedback*). Output is delivered either through projection onto the workpiece (*in situ*), or through repurposing a builder's smartphone or tablet as an auxiliary display (*decoupled*). Our system is composed of a drill which reports orientation and depth through projected feedback, a compound miter saw which measures steadiness and angle with tablet-based output, and an off-the-shelf digital distance measurement tool. Tool actions are coordinated by a central server that maintains the macro's model, tracks its current assembly status, and receives and updates measurements of model parameters (e.g., desired width, height, depth). Macros also generate paper templates to assist novices with alignment and engraving.

Skill-building Activities

Skill-building activities are standalone Processing programs which guide the user through performing repetitions of a specific tool operation. Each repetition, or level, may be guided or unguided, displaying feedback at different times via the tool's display (Figure 9). During guided levels, *continuous, corrective* feedback is displayed to the user. In an unguided level, feedback is instead presented *after* the task is complete.

The drill has two skill-building activities, one to build technique for drilling at specific angles (most commonly 90° and

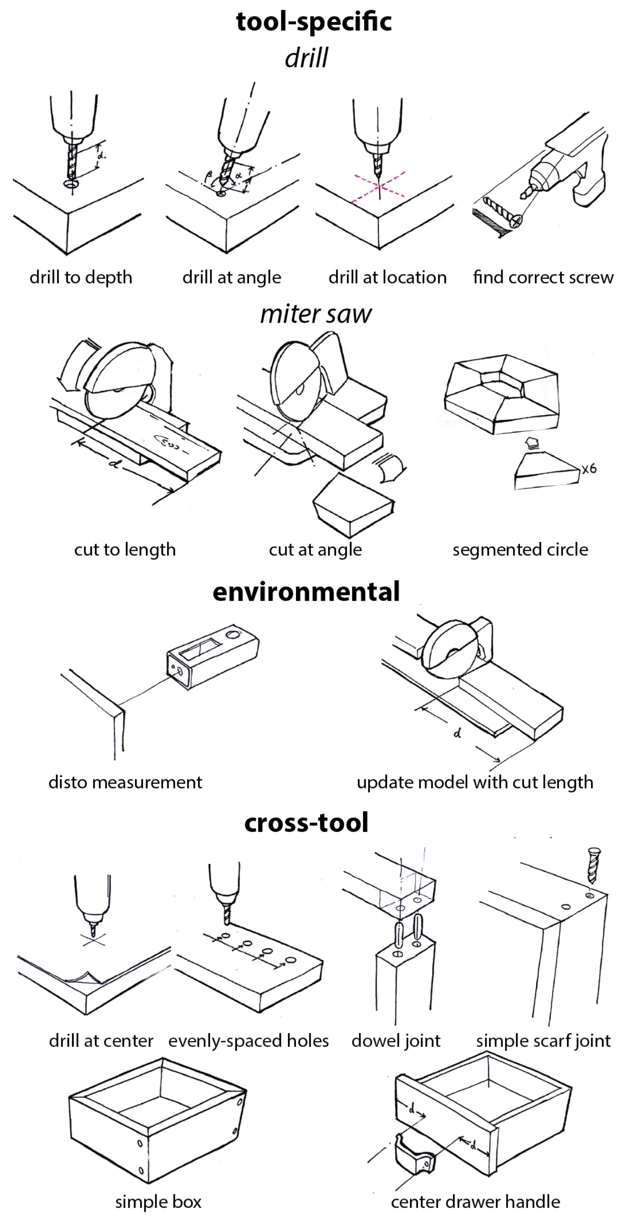

Figure 5. Assembly macros consist of various tool-specific, environmental, and cross-tool operations.

45°) and another to develop intuition for drilling or driving to a certain depth. In these two activities, feedback for both angle and depth are displayed during task execution in guided levels. In unguided levels, real-time feedback for the activity's corresponding parameter (depth or angle) is removed. Following task completion, users are instead presented with their average angle or final depth. The activity formats for both angle and depth are static and consist of 10 levels total, beginning with 3 guided levels, an alternation of unguided and guided levels, and 3 unguided levels, followed by an overall summary of statistics.

The miter saw's skill-building activity trains a user to maintain a steady stroke while making cuts. In contrast to the drill's training activities, all repetitions are unguided to allow complete attention on the saw during its operation. The user is provided a plot of the saw's position over time, and a score of how closely they matched the ideal linear curve.

Assembly Macros

Assembly macros consist of instructions and parameters to describe common construction tasks spanning the capabilities of our tools (see Figure 5). *Tool-specific* macros encode degrees of freedom for machining operations intrinsic to a particular tool, such as drilling a hole to a particular depth or angle. *Environmental* macros incorporate measurements from a user's environment and preceding actions into a digital assembly model, such as cutting a board to the desired width of a bookcase. *Cross-tool* macros utilize environmental information to inform operations using multiple tools, describing features common to traditional woodworking practice, such as joinery. All macros may be used atomically, to aid a user in a free-form project, or be chained together to create complete, parametrized assemblies.

Instruction File Format

We developed a JSON-based domain-specific language to express customizable macros as a sequence of fabrication and assembly steps. Steps are stored in an ordered list, recording `tool` and `task` (e.g. drilling to specific depths, cutting a board to length), for all steps, alongside task-specific parameters such as `x`, `y`, `depth`, `z_orientation`, etc., (see Figure 6, left). These parameters may be static or dynamic. Dynamic parameters may be expressed as formula-based constraints, e.g. `"x":"(keyrack_width - 2*margins)/ number_screws"` for calculating even spacing for screws. Such constraints allow for environmental and cross-tool macros to update the assembly model and the following dependent steps.

Paper Templates

To generate paper templates, macro instructions provide a standalone Processing program with a list of arguments of desired edges, drill locations, vector drawings, and their corresponding positions in inches. The program then draws these features into their proper locations, representing edges as lines, drill locations as 3mm circles, while vector drawings are unmodified and retain their scale. These templates are rendered as absolute-sized PDFs and automatically sent to a HP M451dn color laser printer through the Line Printer Remote protocol [4]. Builders retrieve, align, and position the template to the workpiece according to instructions printed directly on the template. Once the template is aligned, users can drill at the indicated fiducial marks or engrave designs by following vector drawings with a dremel, allowing for cross-tool interactions (see Figure 6, right).

```
{
  "dynamic_params": ["x", "y"],
  "tool": "drill",
  "task": "locateGuides",
  "x": "(keyrack_width - 2*margins)
              / number_screws",
  "y": "material_thickness / 2",
  "z_orientation": 70.0
},
{
  "tool": "drill",
  "task": "drillToDepth",
  "dynamic_params": ["depth"],
  "depth": "screw_depth"
},
```

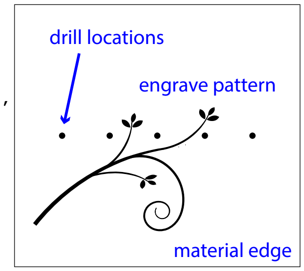

Figure 6. (Left) Our JSON-based language has dynamic and static parameters for instructions; formulas are evaluated in Python. This language captures all tasks, like drilling at a particular location and angle (top) to a specified depth (bottom). (Right) Printable paper templates generated by our system mark locations to drill, patterns to engrave, and material edges for alignment.

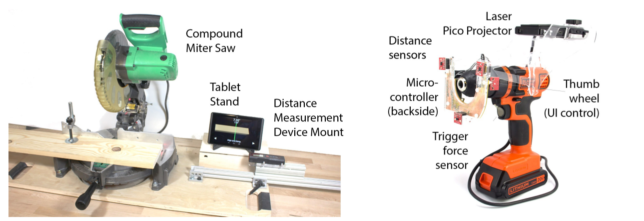

Figure 7. The compound miter saw setup (left) includes the saw itself, instrumented with two rotary potentiometers to determine angle and blade position, a laser rangefinder mounted at a known distance, and a tablet for information display. Our augmented drill has a series of distance sensors and a microcontroller to determine pose, a FSR on the trigger to detect activation, as well as a laser picoprojector for output (right).

Server/Proxy

The server provides three major functions: (1) maintaining macro progress and parameters, (2) modifying the macro parameters from user input, and (3) forwarding current step information to the proper tools. Tools poll the server at regular intervals to retrieve information. Once a tool reports the completion of a step, actual measurements are recorded and the server updates all subsequent steps which depend on that measurement.

Tools

The essence of each augmented power tool is a commodity tool augmented with sensors and/or a display. Individual smart tools have wireless radios to communicate with the server.

Drill

The drill is an instance of a handheld tool in a "gun" configuration with a prehensile, prismatic grasp of the hand around the tool [50], where orientation and distance with respect to the workpiece are underconstrained, i.e., left up to the user to control. Other tools with similar characteristics that could

[4] http://www.cl.cam.ac.uk/cgi-bin/manpage?lpr

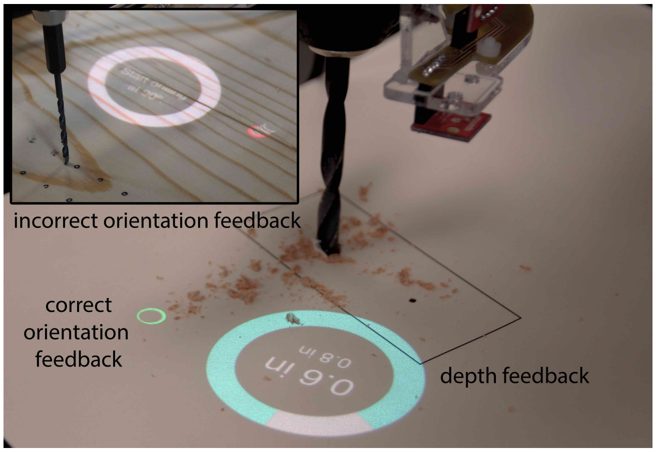

Figure 8. To drill a hole, the user aligns to fiducials on a paper template. They are given feedback on drill perpendicularity, and a progress bar for depth. The drill's rangefinder array keeps display size constant.

benefit from augmentation include nail guns, impact drivers, and handheld reciprocating saws.

The augmented power drill is a retrofitted handheld Black & Decker LDX120C Cordless Drill (see Figure 7, right). The drill can sense its distance and orientation with respect to a surface, and whether or not a builder is activating the drill.

For orientation, we leverage an array of three VL6180X time-of-flight range finders, arranged in a triangle formation, to reveal the drill's relative plane with respect to the workpiece. Using the known distances between the sensors and their measured distances, the drill's pitch and yaw are calculated as follows, where d_{xy} is the distance between sensors x and y, and d_x is the distance from the workpiece reported by sensor x:

$$\text{pitch} = atan(\frac{d_2 - d_3}{d_{23}})$$
$$\text{yaw} = atan(\frac{d_1 - (d_2 + d_3)/2}{sqrt(d_{12}^2 + d_{23}^2)})$$

These sensors are also used to sense depth, for example to ensure that a screw will sit flush in the workpiece. The drill's distance from the workpiece is simply the average of the distances reported by the range finders. Depth is then determined by calculating the difference in the drill's distance between two positions.

Drill activation is sensed with a force sensing resistor attached to the drill's trigger and a photocell mounted on the drill's LED work light (the LED turns on only when the drill's spindle is turning). These sensors are treated as binary indicators for whether the user is executing an operation, and when that operation begins and ends.

Images projected from the top-mounted MicroVision SHOWWX+ pico projector are corrected for size and perspective distortion, ensuring that the device always projects a UI with the same size and aspect ratio. This requires multiplying the display image, or the scene, by a scaling matrix proportional to the drill's distance from the workpiece to maintain constant size. This is then multiplied by the drill's extrinsic matrix, which represents the orientation of the drill relative to the scene. This is obtained by using the yaw and pitch derived above, and applying these values to a rotation matrix. To account for the projector's properties, such as keystone correction[5], the intrinsic matrix of the projector must be determined. This can be done using tools such as ProCamCalib[6], which require analysis of the projection at various angles.

The current prototype requires a manual tether to a laptop to generate graphics, though an on-board computer such as a Raspberry Pi could remove this restriction. The laptop is also used to provide WiFi communication between the drill and central server through a Processing network client. The drill's sensor data is sent to the laptop via serial communication.

The drill supports the following four tasks:

Finding a location to drill: Using its perspective-corrected projection and ability to display absolute-sized distances, the drill can help the user locate where to drill by projecting crosshair guidelines. Drill Sergeant guides the user to manually align these to previously drilled holes, or the natural geometry (corners/edges) of the workpiece. With the alignment complete, the bit is in the correct position for drilling.

Ensuring drill angle: Drill pose is sensed using the rangefinder array, with the drill's pitch and yaw determined using trigonometry. The drill displays a cylinder that mirrors the drill's bit and tilts from the perspective of the user in the same way the drill is tilted. Arrows beside the cylinder show users the correct direction to tilt the drill to adjust for orientation. The cylinder highlights in red when the angle is incorrect, and green when it is right (see Figure 9)

Guiding drill depth: When a builder presses the trigger to activate drilling, the rangefinders track the change in distance between the current position and the user's initial position. Drill depth is displayed as a blue progress bar filling towards the target. Once the user reaches the desired depth, the progress bar turns green to indicate completion. However, if the user continues drilling past the target depth, the progress bar turns red. Throughout, the user is also provided with a readout of their current depth.

Finding the right screw: Finding a screw of the correct length is crucial to ensure an assembly is tightly fastened without exposing screw tips. The drill projects a constant-sized highlight (corrected for pose, as above) of the necessary screw length to help find an appropriate screw in a pile. In this highlight, the screw itself is colored green and indicates where the user must align their screw. The body of the projected screw is enclosed in a red rectangle for easy verification of the screw's length. If the end of the actual screw contains red highlight, the screw is too long. If the red highlight is apparent beyond the screw's length, the screw is too short. Similarly, a rectangular yellow highlight serves as a visual aid for the screw's head.

Distance Measurement

[5] http://www.projectorpeople.com/resources/keystone-correction.asp

[6] http://www.ok.ctrl.titech.ac.jp/res/PCS/research/procamcalib/

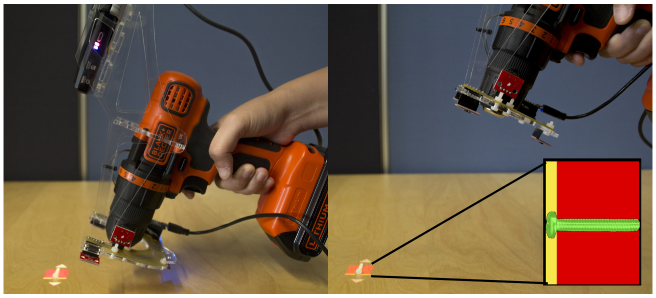

Figure 9. To aid in finding screws, the drill applies perspective correction such that an image's size and aspect-ratio is maintained regardless of orientation. A colored highlight (bottom right) is used to easily identify screws.

The laser distance finder is an example of a measurement device which can be used to precisely determine material size or mark exact positions. Similar tools are tape measures, rulers, and calipers.

We use an off-the-shelf Leica Disto E7100i laser distance finder, which can stream continuous distance measurements accurate to within ± 1.5 mm (± 0.06") over Bluetooth Low Energy at 3Hz. These measurements are received by a tablet and then forwarded to our server via WiFi. This device can be used for free-hand measurements as well as mounted onto other tools in specific locations to aid with measurement tasks relative to another machine tool.

Compound Miter Saw

The compound miter saw is an instance of a stationary power tool where material is inserted into the tool to be processed. Other tools that share these characteristics are table saws, non-CNC router tables, drill presses, and band saws.

Our saw is a Hitachi C10FCH2 10-inch single bevel compound miter saw. A tablet is mounted next to the body to display measurement readouts and instructions (see Figure 7, left).

To measure material length for cutting, the Leica laser distance finder is mounted at a known distance from the blade. Given that d_b is the known distance from the distance finder to the blade, and d_m is measured distance of the material to the distance finder, the cut length will be $d_b - d_m$. Miter angle is determined with a rotary potentiometer mounted on the saw's underside, accurate to $\pm 0.5°$. The blade's position is sensed by a rotary potentiometer, which is used to gauge steadiness. Data for cut length, angle, and blade position is sent via Bluetooth Low Energy to the tablet, which communicates to the server through WiFi.

The saw assists with the following three tasks:

Tracking stroke steadiness: The position of the saw's blade is tracked by a rotary potentiometer attached to the saw's arm. Blade position is recorded in even intervals during the blade's descent, until it reaches the very bottom. To maintain the builder's focus on the saw while cutting, feedback is not displayed until the blade is returned to its initial position. The feedback provides a graph of the blade's position over time in red. This is plotted against the ideal linear curve in green, representing the case of constant velocity throughout the entire downstroke. The user is also given a score calculated using the mean absolute error between the two curves.

Ensuring miter angle: Miter angle is sensed using a rotary potentiometer on the saw's underside. The tablet displays the saw's current angle and the desired angle on a skeuomorphic miter angle readout. When the saw is set at the correct angle, the indicator changes from red to green.

Guiding cut length: The distance ranger points down the length of a board inserted for cutting, such that it can tell where the saw will cut the board. The mounted tablet display helps refine positioning before a cut, displaying an image of a board whose sliding movements reflect those of the user's own board. A solid line on the virtual board indicates where the board will be cut at its current position, and a dotted line shows where it should be cut to achieve the desired length. When aligned, these lines turn green. The display also provides left and right arrows that show which direction the builder should move the board to cut at the correct length, along with a precise digital readout of the current cut length.

EVALUATION

We explored the effects of our training activities on confidence and technique through a formal evaluation with 17 novice users. During this evaluation, we also gauged our assembly macros' ability to assist with the building process. In order to validate the utility, usability, and flexibility of Drill Sergeant, we built several example objects by chaining together construction macros.

Evaluation with Novice Users

In order to gain more insight into how Drill Sergeant may help novices develop their confidence and skills with power tools, we conducted a user evaluation with 17 novices from our institution's fabrication labs. Our users were 18–25 years old (9 male) and had previous basic safety training for handling power tools, but little to no hands-on experience. Sessions lasted less than 1.5 hours each.

Method

Drill Task. Participants were randomly split into an augmented group (8) and a control group (7): all participants used our augmented drill to control for accessory weight, but the control group had projection turned off. Each user was asked to drill two sets of seven holes, at 90° and 45°. Users in the treatment group were provided with the drill's real-time feedback for orientation during the first four holes. Then, to test for development of tacit knowledge, these users were only given post-feedback for the final three holes. Users in the control group were supplied with a protractor and thin rod such that they could manually measure angle at their own discretion after any trial. To gauge improvement, we manually calculated deviation from the target angle for all user trials.

Saw Task. Users were also asked to cut seven 1" blocks of wood as steadily as possible with the miter saw. Participants in the first group were video recorded, then shown the video so they could gauge their performance after each cut. The second group was provided explicit tablet feedback of their steadiness after completing each cut. The augmented and

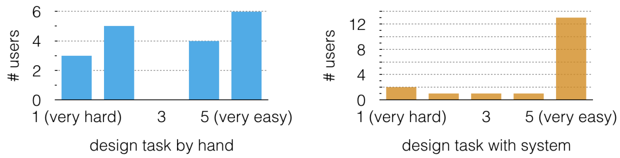

Figure 10. All users self-reported the challenge posed by our design task in the unaugmented (left) and augmented (right) conditions. The bimodal distribution in the "by hand" graph indicates some users had experience with engineering or other design tasks, but all users found our system easy to use.

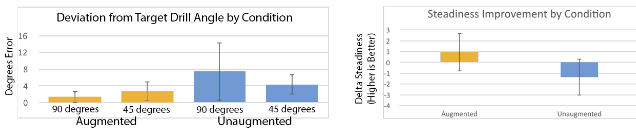

Figure 11. Users in the augmented and unaugmented drill groups had statistically indistinguishable performance (left). The users in our augmented saw condition improved over time, while unaugmented saw users got worse (right).

unaugmented groups were swapped from the drill trials. We measured overall steadiness as the sum of the error (compared to the ideal curve) over all measured positions in each cut.

Box Building Task. Finally, users in both groups executed a design task, both with and without our system. They were asked to create a box frame with an outer perimeter of exactly 18”(457.2*mm*), with the specific dimensions of the box’s length and width left up to the user. To do this without our system they were provided with a measuring tape, paper, and pen. Our box macro provided augmented assistance, allowing the user to make an initial cut to define the box’s length, then guiding them through the remaining cuts.

Prior to beginning these tasks, users were asked to rate their comfort level for performing each operation. Immediately following the first and last operations, they were again asked to rate their comfort level for performing the task. We also collected unstructured feedback on what made each task hard or easy.

Results

As expected, users in both the augmented and unaugmented tool groups reported increased confidence on all tasks following repetitive performance, with no significant difference between conditions.

Participants had comparable accuracy in the drilling task. We did not see statistically significant differences between groups for any drill task. For our statistics, we count angular deviations of $\pm 5°$ from the target as “on target”, and measure deviation beyond that. In the augmented condition drilling perpendicularly, users were off by $\mu = 1.3°, \sigma = 1.4°$ over all trials, and in the 45° condition they were off by $\mu = 7.6°, \sigma = 14.4°$. Users in the unaugmented condition were off from 90° by $\mu = 2.6°, \sigma = 2.3°$ and 45° by $\mu = 4.4°, \sigma = 2.4°$ (see Figure 11, left). This may be due to the limited number of trials and lack of time for users to mentally practice or internalize this tacit knowledge[35]. Many users’ performance also degraded slightly towards the end; one user noted that *“after continuous tasks my arm got a bit tired and it was harder to keep a steady hand.”*

However, users in the augmented condition reported several perceived benefits, saying *“[the drill] gave me more confidence that I was drilling properly. The measuring system really made it easy and enjoyable,”* while users drilling by hand explained that, *“making sure the angle is right can be difficult without a reference.”*

Participants who used the augmented saw were steadier in their cuts. In the augmented group we saw average steadiness improvement of $\mu = 4.64\%, \sigma = 8.66\%$, while in the unaugmented group we saw $\mu = -6.71\%, \sigma = 8.37\%$ (see Figure 11, right). While our augmented users improved in steadiness, users who only watched their videos actually got worse! As repeatedly using the saw resulted in increased confidence, many users in the unaugmented group stopped observing their video feedback. Meanwhile, at minimum, users presented with the graph quickly assessed their feedback following each cut.

Participants who used augmented tools produced more accurate boxes in our design exercise than participants in the control condition. We manually measured the perimeters of the boxes generated by users and calculated absolute value of error compared to the target of 18”. Boxes created by hand had an average error of 1.21” $\pm$ 1.11”, while the average over users (after removing an outlier who disregarded the provided instructions) using the macro was 0.08” $\pm$ 0.06”. Including the outlier, the average error for users leveraging Drill Sergeant was 0.24” $\pm$ 0.67”. Even a task as simple as box building has subtle design decisions—e.g., how do you account for the material’s thickness? Participants in the control condition were confounded by these decisions.

Qualitative Observations

Users were overall very positive about using our system. One remarked that *“with the system people like me can start working on wood without supervision and the tools provide encouragement to take projects.”* Another said *“[the augmented tool feedback is] fairly good training in some ways, and helps show that these tools aren’t actually that scary. They’re mostly just loud I guess.”* When asked what projects they would like to use augmented tools for, our users described a variety of home improvement and craft tasks, as well as one user who mentioned a side benefit of our system: *“A sculpture of an abstract looking house. Even though it didn’t need to be precise it would have saved me money buying and not wasting wood.”*

On the particular tasks we tested, all users in the augmented conditions likewise reported positive benefits: *“sensor that told me what to do assured [sic] me,”* however 5 users noted the *“bulkiness of the extra accessories”* on the drill, especially in the 45° condition where the *“heavier drill makes holding one angle more complicated.”* This could be remedied with additional industrial design. The miter saw’s augmentations do not interfere with tasks in the same way; none of our users directly commented on the design of the tool itself, but instead focused on the design of the information feedback and the actual cutting task.

Example Objects

Figure 12. As part of our validation, we created five example objects showcasing our tools' cooperation: two parameterized birdhouses (left), a mount for a network switch (center left), a box using our frame macro (center right), and a chair with legs matching the user's height (right).

Five building projects utilize our ecosystem of compound miter saw with display, smart drill, digital distance measurement, and paper templates. They are created by chaining together various tool-specific, environmental, and cross-tool macros.

Birdhouses

A birdhouse is an example of a typical beginner's woodworking project. Our birdhouses leverage two differently-sized materials—1x4s for the sides and 1x6s for the roof, front, and back. Each birdhouse was designed by chaining together environmental, cross-tool, and tool-specific macros to indicate the birdhouse's desired dimensions and aid in precise machining (e.g. cutting the correct angle and length, drilling at correct positions and depths). Paper templates are used to help engrave a window on the house's front panel, and indicate drilling positions for screws and the entryway. These two projects demonstrate the ease of rescaling a model to users' desired measurements based on our parameterized file format. Some macros utilized during the process require reinserting a single block of wood into the miter saw multiple times. For instance, the front piece of one birdhouse has two 45° cuts meeting at the roof peak, in addition to a 90° sizing cut at the bottom. This points towards the possibility of creating more complex joinery involving multiple cycles with our augmented tools.

Switch Mount

The switch mount consists of two screws and a single 1x6, which represents a surface where a network switch may be seated. For a proper fit, the two screws must be at a precise height and relative distance to one another. The mount utilizes an environmental macro to input the position of the second screw relative to the first, which informs the drill's guide alignment macro, indicating position (see Figure 13). To assist in the actual drilling process, the mount leverages tool-specific macros for depth and angle.

Box

The box is made of six pieces of 1x4 lumber with 45° beveled edges and is generated using our simple box cross-tool macro. Its width can be freely defined by users, but the depth is constrained by the material width of boards serving as the base In addition to the augmented miter saw and measurement device, the box construction macro uses the drill's projected alignment guides and orientation feedback for drilling and driving screws through the corners to secure the frame. These screws must be placed within 0.5" from each edge so they do not protrude into the interior of the box.

Chair

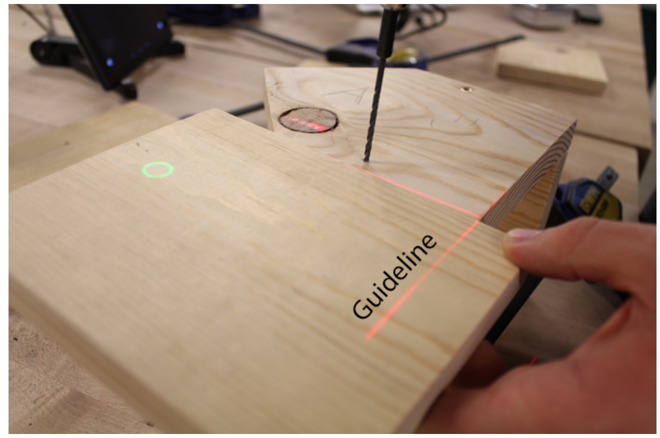

Figure 13. To locate drill positions, the user aligns the projected crosshair guidelines to previously drilled holes or the workpiece's natural geometry.

We constructed a custom height chair from 2x4 lumber pieces. Most chair plans have pre-defined dimensions for fitting 95% of the population, but do not fit those who are very short or tall[44]. To create a chair with Drill Sergeant, a user utilizes the "disto measurement" environmental macro and measures the length of their leg below the knee to define the length of the chair legs, thereby scaling the chair in height. The saw's tool-specific "cut to length" macro assists in making uniform cuts for the chair's legs, seat, and back, which are then fastened using the drill's tool-specific macros for depth and angle.

LIMITATIONS AND FUTURE WORK

Drill Sergeant has several limitations – some are inherent to the approach, others are shortcomings of our current implementation.

No holistic tracking of user's success: Drill Sergeant can only display feedback for its instrumented tools. We cannot determine if a user performs other tasks, like clamping or sanding, incorrectly, though errors in such steps can lead to project failure.

More tool characteristics could be instrumented: Higher fidelity sensing of tools combined with machine learning may allow for more feedback on performance and use, for example, monitoring pressure and auditory or EM signatures of drill performance to suggest proper force and speed.

Limitations from hardware and sensing of tools restrict our system to larger, rectilinear source material. For instance, orientation sensing on the drill depends on all three distance sensors hitting the same plane. Similarly, projecting feedback assumes that there is a planar surface to project on just above the drill point.

Exploring long-term benefits of tool training: Our evaluation studies the effects of skill-building activities after a series of short trials. This does not measure our system's capacity to make long-term improvements to a user's tool technique, which would require measuring users' skills over extended periods of time.

Adding new augmented tools: During our evaluation, users noted that other augmented tools could have greatly assisted with a personal project. The augmentation of new tools would allow us to further explore the tool augmentation design space. One potential domain is finishing tools (e.g., grinders, sanders) that can help users achieve an even coat.

CONCLUSION

Dynamic software tutorials have helped countless people bootstrap themselves into new skills and improve their grasp of existing ones. With Drill Sergeant, we extend the ideas of these tutorials into the physical realm, leveraging an ecosystem of augmented tools that work together to track and give real-time feedback to a user as they develop their technique and build construction macros. Our evaluation with novice users suggest that the interaction between smart tools and real-time feedback has potential in helping novices build their confidence and skills with power tools. As with the receding feedback of our skill-building activities, we believe augmentations for tool-centric skills is scaffolding that could fade over time [41], so users become proficient even without augmentations. Meanwhile, results from our evaluation and example objects also show promise in our system's ability to assist with executing construction macros and using those macros to create complete, customizable projects. In contrast to feedback for tool-centric skills, project-specific assembly instructions and measurements that change from project to project and should always be delivered through Drill Sergeant. Looking forward, we envision sensors integrated into all available tools, and software systems supporting the interactions described in this paper as easy-to-use downloadable applications. Everyone from Bob Villa to a weekend-warrior should be able to quickly and easily execute woodworking and other DIY projects.

ACKNOWLEDGMENTS

This work was supported in part by TerraSwarm, one of six centers of STARnet, a Semiconductor Research Corporation program sponsored by MARCO and DARPA. This material is also based on work supported by the NSF under Grant No. DGE 1106400.

REFERENCES

1. Agrawal, H., Umapathi, U., Kovacs, R., Johannes, F., Chen, H.-T., Mueller, S., and Baudisch, P. Protopiper: Physically sketching room-sized objects at actual scale. In *Proceedings of UIST'15*, To appear.
2. Antifakos, S., Michahelles, F., and Schiele, B. Proactive instructions for furniture assembly. In *Proceedings of the 4th International Conference on Ubiquitous Computing*, UbiComp '02. Springer-Verlag, London, UK, UK, 2002, 351–360.
3. Bergman, L., Castelli, V., Lau, T., and Oblinger, D. Docwizards: A system for authoring follow-me documentation wizards. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology*, UIST '05, ACM (New York, NY, USA, 2005), 191–200.
4. Berthouzoz, F., Li, W., Dontcheva, M., and Agrawala, M. A framework for content-adaptive photo manipulation macros: Application to face, landscape, and global manipulations. *ACM Trans. Graph. 30*, 5 (Oct. 2011), 120:1–120:14.
5. Campbell, T., Harper, J., Hartmann, B., and Paulos, E. Towards digital apprenticeship: Wearable activity recognition in the workshop setting. Tech. Rep. UCB/EECS-2015-172, EECS Department, University of California, Berkeley, Jul 2015.
6. Chi, P.-Y., Ahn, S., Ren, A., Dontcheva, M., Li, W., and Hartmann, B. MixT: Automatic generation of step-by-step mixed media tutorials. UIST '12, 93–102.
7. Dong, T., Dontcheva, M., Joseph, D., Karahalios, K., Newman, M., and Ackerman, M. Discovery-based games for learning software. CHI '12, 2083–2086.
8. Fishkin, K. P. A taxonomy for and analysis of tangible interfaces. *Personal Ubiquitous Comput. 8*, 5 (Sept. 2004), 347–358.
9. Follmer, S., Carr, D., Lovell, E., and Ishii, H. CopyCAD: Remixing physical objects with copy and paste from the real world. UIST '10 Adjunct, 381–382.
10. Grabler, F., Agrawala, M., Li, W., Dontcheva, M., and Igarashi, T. Generating photo manipulation tutorials by demonstration. 66:1–66:9.
11. Grossman, T., Matejka, J., and Fitzmaurice, G. Chronicle: Capture, exploration, and playback of document workflow histories. In *Proceedings of the 23Nd Annual ACM Symposium on User Interface Software and Technology*, UIST '10, ACM (New York, NY, USA, 2010), 143–152.
12. Gupta, A., Fox, D., Curless, B., and Cohen, M. Duplotrack: A real-time system for authoring and guiding duplo block assembly. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*, UIST '12, ACM (New York, NY, USA, 2012), 389–402.
13. Henderson, S., and Feiner, S. Exploring the benefits of augmented reality documentation for maintenance and repair. *Visualization and Computer Graphics, IEEE Transactions on 17*, 10 (Oct 2011), 1355–1368.
14. Hooper, C. J., Preston, A., Balaam, M., Seedhouse, P., Jackson, D., Pham, C., Ladha, C., Ladha, K., Plötz, T., and Olivier, P. The french kitchen: Task-based learning in an instrumented kitchen. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, UbiComp '12, ACM (New York, NY, USA, 2012), 193–202.
15. Ju, W., Hurwitz, R., Judd, T., and Lee, B. Counteractive: An interactive cookbook for the kitchen counter. CHI EA '01, 269–270.
16. Knabe, K. Apple Guide: A Case Study in User-aided Design of Online Help. CHI '95, 286–287.
17. Knibbe, J., Grossman, T., and Fitzmaurice, G. 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, 2015), 83–92.

18. Kong, N., Grossman, T., Hartmann, B., Agrawala, M., and Fitzmaurice, G. Delta: A tool for representing and comparing workflows. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, ACM (New York, NY, USA, 2012), 1027–1036.
19. Kranz, M., MöLler, A., Hammerla, N., Diewald, S., PlöTz, T., Olivier, P., and Roalter, L. The mobile fitness coach: Towards individualized skill assessment using personalized mobile devices. *Pervasive Mob. Comput. 9*, 2 (Apr. 2013), 203–215.
20. Laput, G., Adar, E., Dontcheva, M., and Li, W. Tutorial-based interfaces for cloud-enabled applications. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*, UIST '12, ACM (New York, NY, USA, 2012), 113–122.
21. Laput, G., Yang, C., Xiao, R., Sample, A., and Harrison, C. Em-sense: Touch recognition of uninstrumented, electrical and electromechanical objects. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, UIST '15, ACM (New York, NY, USA, 2015), 157–166.
22. Lee, J., Su, V., Ren, S., and Ishii, H. HandSCAPE: A vectorizing tape measure for on-site measuring applications. CHI '00, 137–144.
23. Linehan, C., Kirman, B., Lawson, S., and Chan, G. Practical, appropriate, empirically-validated guidelines for designing educational games. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, ACM (New York, NY, USA, 2011), 1979–1988.
24. Lukowicz, P., Ward, J., Junker, H., StÃď'ger, M., TrÃűster, G., Atrash, A., and Starner, T. Recognizing workshop activity using body worn microphones and accelerometers. In *Pervasive Computing*, A. Ferscha and F. Mattern, Eds., vol. 3001 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2004, 18–32.
25. Marshall, J. Smartphone sensing for distributed swim stroke coaching and research. In *Proceedings of the 2013 ACM Conference on Pervasive and Ubiquitous Computing Adjunct Publication*, UbiComp '13 Adjunct, ACM (New York, NY, USA, 2013), 1413–1416.
26. Mohr, P., Kerbl, B., Donoser, M., Schmalstieg, D., and Kalkofen, D. Retargeting technical documentation to augmented reality. CHI '15, ACM (2015), 3337–3346.
27. Mueller, S., Lopes, P., and Baudisch, P. Interactive construction: Interactive fabrication of functional mechanical devices. UIST '12, 599–606.
28. Nelson, J., and Nelson, J. *The Big Book of Weekend Woodworking: 150 Easy Projects*. Lark Crafts, 2005.
29. Oda, O., Elvezio, C., Sukan, M., Feiner, S., and Tversky, B. Virtual replicas for remote assistance in virtual and augmented reality. To appear.
30. Olwal, A., Gustafsson, J., and Lindfors, C. Spatial augmented reality on industrial CNC-machines. vol. 6804 (2008), 680409–680409–9.
31. Pérez, J. D. G., Payá, A. S., Fernández, D. R., Sánchez, S. H., and Alonso, O. M. Ubiquitous low-cost sports training system for athletes. In *Proceedings of the 6th Euro American Conference on Telematics and Information Systems*, EATIS '12, ACM (New York, NY, USA, 2012), 105–112.
32. Polanyi, M. *The Tacit Dimension*. University of Chicago Press, Chicago London, 1966, 2009.
33. Rivers, A., Adams, A., and Durand, F. Sculpting by numbers. *ACM Trans. Graph. 31*, 6 (Nov. 2012), 157:1–157:7.
34. Rivers, A., Moyer, I. E., and Durand, F. Position-correcting tools for 2D digital fabrication. *ACM Trans. Graph. 31*, 4 (July 2012), 88:1–88:7.
35. Rogers, R. G. Mental practice and acquisition of motor skills: Examples from sports training and surgical education. *Obstetrics and Gynecology Clinics of North America 33*, 2 (2006), 297 – 304. Teaching and Evaluating Surgical Skills.
36. Ryobi Inc. Phoneworks smart phone tools. https://www.ryobitools.com/phoneworks/, 2015.
37. Sato, A., Watanabe, K., and Rekimoto, J. Mimicook: A cooking assistant system with situated guidance. In *Proceedings of the 8th International Conference on Tangible, Embedded and Embodied Interaction*, TEI '14, ACM (New York, NY, USA, 2013), 121–124.
38. Schmidt, P. *PlyDesign: 73 Distinctive DIY Projects in Plywood*. Storey Publishing, 2012.
39. Schneegass, S., Sahami Shirazi, A., Döring, T., Schmid, D., and Schmidt, A. NatCut: An interactive tangible editor for physical object fabrication. CHI '14 EA, 1441–1446.
40. Shaper Tools. Origin smart cutting tool. http://shapertools.com, 2015.
41. Soloway, E., Guzdial, M., and Hay, K. E. Learner-centered design: The challenge for hci in the 21st century. *interactions 1*, 2 (Apr. 1994), 36–48.
42. Song, H., Grossman, T., Fitzmaurice, G., Guimbretiere, F., Khan, A., Attar, R., and Kurtenbach, G. PenLight: Combining a mobile projector and a digital pen for dynamic visual overlay. CHI '09, 143–152.
43. Spina, G., Huang, G., Vaes, A., Spruit, M., and Amft, O. Copdtrainer: A smartphone-based motion rehabilitation training system with real-time acoustic feedback. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '13, ACM (New York, NY, USA, 2013), 597–606.

44. Tilley, A. R., and Wilcox, S. B. *The measure of man and woman : human factors in design*. Wiley, New York, 2002.

45. Torrey, C., Churchill, E. F., and McDonald, D. W. Learning how: The search for craft knowledge on the internet. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, ACM (New York, NY, USA, 2009), 1371–1380.

46. Wakkary, R., Schilling, M. L., Dalton, M. A., Hauser, S., Desjardins, A., Zhang, X., and Lin, H. W. Tutorial authorship and hybrid designers: The joy (and frustration) of diy tutorials. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, ACM (New York, NY, USA, 2015), 609–618.

47. Weichel, C., Alexander, J., Karnik, A., and Gellersen, H. SPATA: Spatio-tangible tools for fabrication-aware design. TEI '15, 189–196.

48. Weichel, C., Lau, M., Kim, D., Villar, N., and Gellersen, H. W. MixFab: A mixed-reality environment for personal fabrication. CHI '14, 3855–3864.

49. Willis, K. D., Xu, C., Wu, K.-J., Levin, G., and Gross, M. D. Interactive fabrication: New interfaces for digital fabrication. TEI '11, 69–72.

50. Zheng, J., De La Rosa, S., and Dollar, A. An investigation of grasp type and frequency in daily household and machine shop tasks. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on* (May 2011), 4169–4175.

51. Zoran, A., and Paradiso, J. A. FreeD: A freehand digital sculpting tool. CHI '13, 2613–2616.

52. Zoran, A., Shilkrot, R., Goyal, P., Maes, P., and Paradiso, J. The wise chisel: The rise of the smart handheld tool. *Pervasive Computing, IEEE 13*, 3 (July 2014), 48–57.

53. Zoran, A., Shilkrot, R., Nanyakkara, S., and Paradiso, J. The hybrid artisans: A case study in smart tools. *ACM Trans. Comput.-Hum. Interact. 21*, 3 (June 2014), 15:1–15:29.