Don't Mesh Around: Streamlining Manual-Digital Fabrication Workflows with Domain-Specific 3D Scanning

Ilan Moyer
Massachusetts Institute of Technology
Cambridge, Massachusetts, USA

Samuelle Bourgault
University of California Santa Barbara
Santa Barbara, California, USA

Devon Frost
University of California Santa Barbara
Santa Barbara, California, USA

Jennifer Jacobs
University of California Santa Barbara
Santa Barbara, California, USA

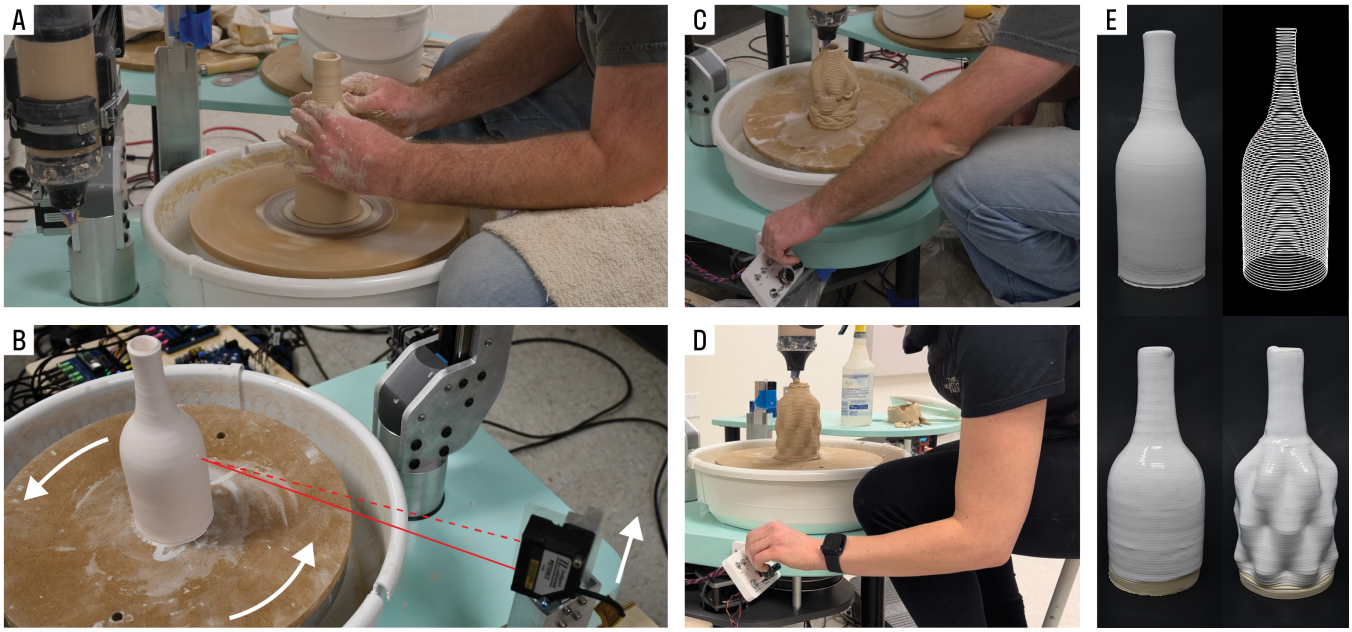

Figure 1: The Craft Assisted Scanner (CAS) is a 3D scanning and printing system that allows practitioners to design for clay 3D printing through manual throwing. A. Practitioners throw vessels on a specialized pottery wheel that is an augmented polar-coordinate 3D printer. B. The CAS records a spiralized toolpath that corresponds with the topology of the thrown vessel through a precision distance sensor that raises on a vertical axis as the wheel spins. The scan can then be immediately printed without leaving the wheel. C-D. An onboard texturization module allows the practitioner to augment the print with surface texture in real time as the recording is played back. E. In clockwise order from the top left, a manually thrown bottle, a visualization of the scanned toolpath, a textured print of the scan, and an untextured print.

ABSTRACT

Software-first digital fabrication workflows are often at odds with material-driven approaches to design. Material-driven design is especially critical in manual ceramics, where the craftsperson shapes the form through hands-on engagement. We present the Craft-Aligned Scanner (CAS), a 3D scanning and clay-3D printing system that enables practitioners to design for digital fabrication through traditional pottery techniques. The CAS augments a pottery wheel that has 3D printing capabilities with a precision distance sensor on a vertically oriented linear axis. By increasing the height of the sensor as the wheel turns, we directly synthesize a 3D spiralized toolpath from the geometry of the object on the wheel, enabling the craftsperson to immediately transition from manual fabrication to 3D printing without leaving the tool. We develop new digital fabrication workflows with CAS to augment scanned forms with functional features and add both procedurally and real-time-generated surface textures. CAS demonstrates how 3D printers can support material-first digital fabrication design without foregoing the expressive possibilities of software-based design.

UIST '24, October 13–16, 2024, Pittsburgh, PA, USA

ACM ISBN 979-8-4007-0628-8/24/10
https://doi.org/10.1145/3654777.3676385

CCS CONCEPTS

• **Human-centered computing → Interaction devices**; *Interaction design process and methods.*

KEYWORDS

digital fabrication, 3D scanning, hardware prototyping, clay 3D printing, craft

ACM Reference Format:
Ilan Moyer, Samuelle Bourgault, Devon Frost, and Jennifer Jacobs. 2024. Don't Mesh Around: Streamlining Manual-Digital Fabrication Workflows with Domain-Specific 3D Scanning. In *The 37th Annual ACM Symposium on User Interface Software and Technology (UIST '24), October 13–16, 2024, Pittsburgh, PA, USA.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3654777.3676385

1 INTRODUCTION

Design in digital fabrication is rooted in software. In the canonical digital fabrication workflow, practitioners begin by specifying solid geometry in general-purpose computer-aided design (CAD) software and then translate that geometry to machine-specific toolpaths in computer-aided manufacturing software (CAM) like a slicer [38]. Only after these processes are completed does the practitioner engage with the physical and material aspects of digital fabrication when they upload the toolpaths to the CNC machine. This CAD-CAM-CNC workflow relegates all explicit design activity to actions within the software, despite the fact that digital fabrication is equal parts digital and material practice. Successful digital fabrication outcomes depend on aligning digitally defined operations with machine constraints and material affordances [19]. In recognition of the limitations of software-first and software-only digital fabrication design processes, we pose the following research question: *How can we relocate digital fabrication design activity to a domain that is familiar and expressive for skilled manual practitioners?*

Clay 3D printing is an ideal context to investigate this question because it relies on identical materials as manual ceramics fabrication while reproducing the software-driven design workflows developed for plastic-based 3D printing [21]. Reproducing CAD-CAM-CNC workflows in clay 3D printing is detrimental in two respects. First, it eliminates the ability of skilled manual ceramics practitioners to leverage their substantial material expertise in designing for clay 3D printing [15]. Second, it reduces the degree to which *any* designer can create digital designs that correspond to the unique material affordances and constraints of clay- namely, its increased plasticity, reduced rigidity, and complex state change in comparison to thermoplastic printing [37]. We see an opportunity to support skilled material design in clay 3D printing by 1) developing *material-first* digital design workflows that begin with the process of hands-on clay manipulation and 2) creating domain-specific digital representations that streamline the transition between digital and material domains.

We present the Craft-Aligned Scanner (CAS), a 3D scanning and printing system that enables practitioners to use throwing on a pottery wheel as a mechanism to create toolpaths for clay 3D printing. Traditional potters create forms by placing clay on a rotational platform (the wheel) and pulling it up by hand to produce radially symmetric vessels of varying shapes and sizes. We augment a specialized pottery wheel (the Digital Pottery Wheel or DPW) that includes a polar-coordinate clay 3D printer [25] with a precision laser distance sensor on a vertical linear actuator. We developed a scanning routine in which we incrementally increase the z-position of the laser as the wheel turns. This enables us to record a spiralized set of values that correspond with the topology of the vessel on the wheel. We use the CAS microcontroller to calculate polar step and direction signals for the axes of the DPW. These signals can be immediately used to drive the DPW printing system to produce a 3D printed object that corresponds with the geometry of a thrown form without any software CAD or CAM intervention by the practitioner.

To our knowledge, CAS is the first integrated 3D scanning and fabrication system for clay 3D printing. Researchers have previously used 3D scanning to reduce challenges in desktop CAD [10, 31] or assist in the process of designing for 3D printing relative to existing physical objects [35, 43]. Our work is different in that we eliminate the requirement for *any* desktop software design operations, although we demonstrate how our system can streamline transitions between CNC use and desktop CAM if desired. Furthermore, prior systems use preexisting depth sensors for volumetric capture, which produces a dense and unstructured point cloud that must be post-processed to function as a toolpath. In contrast, our method produces a continuous spiral toolpath that corresponds with standards for clay 3D printing toolpathing and can be immediately processed by a 3D printer. We make the following contributions:

- A novel 3D scanning hardware device that captures precise object topology by leveraging the rotational properties of a polar-coordinate 3D printer and pottery wheel.
- A 3D scanning algorithm that is compatible with the requirements of clay 3D printing and enables the immediate transition from scanning to printing. Our approach also streamlines modifications of a physical design in parametric CAD and CAM.
- Support for real-time modification of scan printing through a texturizer module that enables the practitioner to introduce controlled oscillations into the motion of the print head during scan playback.
- The demonstration of four new scanning-to-printing workflows that are enabled by a combination of our scanning technology and texturizer. These include 1) print reproduction of a thrown vessel, 2) reproduction of an asymmetrical vessel, 3) integration with desktop CAM, 4) and on-tool texturization of scanned form. We validate these workflows by working with an experienced potter to show how our methods perform when using professionally thrown vessels as input.

2 RELATED WORK

We contrast the CAS with existing methods of 3D scanning for fabrication. We then describe prior research using scanning as a component of digital fabrication systems to reduce barriers and support new design workflows. Finally, we explain how CAS aligns with existing requirements and affordances of clay 3D printing.

2.1 3D Scanning for Additive Fabrication

3D scanning technologies capture dimensional information from physical objects and construct a 3D digital model. This process can allow digital fabrication designers to replicate or reference existing physical objects rather than modeling from scratch in CAD. Common forms of 3D scanning include *structured light*, where a camera captures projected light patterns on the object and geometry is reconstructed in software based on the distortion of light patterns [30], *photogrammetry*: where geometric displacement and object deformation are extracted through multiple photographs taken at different angles [4] and *laser triangulation*: where a laser projects a light pattern on the object which is captured by a sensor (often a digital camera) and object distance is calculated by triangulation [13]. LiDAR scanning also relies on laser range finding but calculates distance using time of flight [41]. CAS uses a laser-triangulation-based distance sensor. Similar to some existing laser-triangulation and structured light platforms, we rotate the object to be scanned. Unlike other laser triangulation systems that either project or raster a line that covers the entire object or rotate the laser and cameras [13], we scan with a single laser point and slowly actuate the emitter along a vertical trajectory. Our approach is uniquely positioned to integrate with the mechanics of polar-coordinate 3D printing systems, thereby supporting a new approach to unified scanner and CNC technology.

Current 3D scanning technologies produce a point cloud that a designer can convert into a polygonal mesh with CAD mesh-processing software [1, 6]. If the mesh is water-tight, the designer can use it for some forms of additive fabrication by converting the mesh to a toolpath with general-purpose slicing software. In many cases, scanning produces artifacts that require mesh repair [3]. Because meshes are unstructured and noisy compared to 3D representations like solid models or NURBS, they significantly reduce the control the designer has over custom toolpath specification. In all cases, for clay 3D printing, practitioners must post-process scan meshes in desktop or mobile CAD and generate toolpaths in CAM software [37] This is also true for most forms of 3D printing [12]. The challenges of CAD and CAM software are well documented [20], and they remove the manual craftsperson from the context of their work. CAS eliminates the need for mesh repair or post-processing because we convert and record scan data as a sequential set of motor step and direction signals that can be replayed to control the axes of a polar-3D printer, *not a mesh*. As a result, CAS scans can be immediately printed or easily parameterized for CAM manipulation.

2.2 Using Scanning to Lower Barriers in Digital Fabrication

In line with prior HCI digital fabrication research, we employ scanning to reduce challenges and introduce new workflows for 3D design for digital fabrication. Our work is generally relevant to interactive fabrication [44] because we contribute a domain-specific method for on-tool scanning and design manipulation.

2.2.1 Supporting Tangible Design Processes. Researchers have used scanning to re-envision or circumvent desktop-based CAD. KidCAD [10] uses structured light scanning to detect deformations in a gel surface [11] to support 2.5D design through manual manipulation, whereas Makers Marks [31] employs 3D scanning of an annotated clay model to automatically generate plastic 3D-printable shells for electronic components. Tactum [16] and ExoSkin [17] use a depth camera to scan a portion of a person's body and track gestures to enable digital design and fabrication, respectively. Kim et al. demonstrate a 3D printing workflow that incorporates real-time input using an RGB camera–including 2D overhead profiles of physical objects–into the 3D printing toolpath through a segmented GCode buffer [22]. Our objective aligns with these prior works in that we circumvent rigid desktop-based design processes for digital fabrication by relocating the design process in clay 3D printing to manual activities like throwing.

2.2.2 Augmenting Existing Physical Objects. 3D scanning can enable designers to use digital fabrication to modify existing artifacts. Retrofab enables designers to scan and annotate a legacy device interface, and the system automatically generates a 3D-printable enclosure [29]. MixFab, an augmented reality CAD system [43], incorporates 3D shape acquisition of physical parts through Kinect-based scanning with gestural manipulation. CustomizAR uses LiDAR-based measurement of physical objects to adapt Thingiverse designs to fit physical objects [23]. With CAS, we augment scans of manually produced forms with CAM-based or on-tool surface textures but do not require screen-based modeling.

2.2.3 Guiding Manual Execution. Scanning can support the learning and execution of skilled manual tasks. Hattab et al. use 3D scanning of an in-progress carved object to visualize differences between the carving and the target model to guide the practitioner [18]. The Robotic Plastering System incorporates LiDAR to adjust a robot arm fabrication trajectory when performing toolpaths over previously fabricated material [24]. Researchers have also used laser scanning in construction to continuously monitor fabrication output and detect errors [26]. We also aim to support skilled manual input in digital fabrication by contributing a domain-specific scanning technique that accurately captures manually thrown clay vessels.

2.2.4 Integrated Scanners and CNC Machines. The CAS functions as an integration of a scanner and a digital fabrication machine and, therefore, contributes to HCI research that combines sensing and fabrication in the same device. Jubilee is a multi-tool CNC machine with automated tool changing that supports fabrication workflows that integrate sensing and fabrication [40]. CopyCAD supports a design-by-example workflow for milling wherein profiles of example objects are captured by webcam and edited on-machine by the practitioner [9]. Other systems seek to support the editing and revision of fabricated artifacts through CNC machines that combine scanning with additive and subtractive end effectors [35, 42]. Sitthi-Amorn et al. developed a multi-material polymer 3D printer with tomography scanning to support print-head calibration and integration of auxiliary components [32]. These prior systems rely on either stationary [9, 35, 42] or carriage-mounted [32, 40] scanners for CNCs with a cartesian mechanism. In contrast, we develop a scanning method that exploits the rotational platform of an existing polar-based 3D printing mechanism to produce a spiral toolpath rather than a point cloud or bitmap image. Furthermore, to our knowledge, CAS is the first integration of a real clay 3D printer and

scanner. The Reform system relies on a polymer-based compound to fabricate clay stand-ins, which are later fabricated on a separate printer in plastic [42]. We fabricate with clay that is identical to materials used in professional ceramics, and our system produces kiln-ready vessels.

2.3 Toolpath Requirements for Clay 3D Printing

Clay 3D printing is the process of extruding wet clay through an end effector at varying spatial positions to produce a 3D vessel [39]. Because clay 3D printing relies on standard clay, resulting artifacts are subject to the same constraints as those in traditional ceramics [37]. Clay 3D printed vessels both allow and often require skilled manual manipulation to produce polished, functional results. The use of clay in additive extrusion introduces additional constraints and affordances not present in thermoplastic printing. Clay 3D printers lack support material and rely exclusively on gravity and layer height for layer lamination, which constrains the amount of overhang of printed vessels in comparison to thrown or coiled forms [14]. Furthermore, most clay 3D extruders lack a retraction mechanism [27] and require significantly larger nozzle diameters than consumer thermoplastic printing (e.g., 1-8mm). Because of these qualities, clay 3D prints are frequently structured as continuous vessels with visible layers. Printing clay vessels with a spiralized non-planar toolpath with an incrementally increasing z-height eliminates visible seams and preserves surface texture. The visible layer structure has advantages. Practitioners can fabricate vessels with unique surface textures and oscillating geometries by rapidly varying the position of the toolpath across the horizontal plane, modulating extrusion rate, and creating portions of unsupported toolpaths [5, 8]. These textures are a key aesthetic affordance of clay 3D printing technology in comparison to manual methods.

Many clay 3D printing workflows are similar to plastic printing, wherein a form is designed in CAD software, sliced, and uploaded to the printer [36]. This workflow can obstruct the unique affordances of clay 3D printing- like surface textures and low-level material control- and makes it challenging for skilled manual ceramics practitioners to integrate manual skill and material expertise in the design process. Domain-specific CAM-based design tools for clay 3D printing like CoilCAM [5] and SketchPath [15] allow practitioners to design at the level of the toolpath.

The Digital Pottery Wheel (DPW) further bridges the gap between clay 3D printing and manual practice by integrating a polar clay 3D printing mechanism with a conventional pottery wheel to directly combine manual throwing and 3D printing in the same machine [25]. CAS significantly extends this prior work. We augment the DPW with a novel scanning device that repurposes the wheel/build platform as a rotational scanning bed. We contribute a scanning algorithm that directly corresponds with the requirements of clay 3D printing. As previously established in section 2.1, existing scanning methods produce an unstructured mesh, and, in all cases, require desktop or mobile CAD post-processing and CAM toolpath generation. In contrast, both CAS and clay 3D printing operate along a spiral path, enabling direct recording and execution of a toolpath without leaving the wheel.

We enable on-tool modification of scanned forms through a texturizer module that enables the practitioner to introduce surface variations. Whereas the prior version of the DPW required software-generated GCode to 3D-print pre-planned forms, the combined CAS features enable practitioners to design 3D-printable forms through throwing or manual sculpting and augment them with precise surface texture in real time. These contributions both extend the capabilities of the DPW and contribute new methods for 3D scanning and printing as a whole.

Our texturizer module has similar principles to the work of Subbaraman and Peek [33]. Our method is different because we generate scanning data as relative step and direction signals that are directly mixed within the DPW real-time motion controller. As a result, our texturizer supports live adjustments with zero latency and no risk of kinematic disruption. In contrast, Subbaraman and Peek modify chunked G-Code commands stored within a buffer and negotiate a tradeoff between increased latency and machine stutter for rapid changes to small machine movements.

3 CAS DESIGN AND IMPLEMENTATION

The CAS system encompasses a custom-built 3D scanning hardware device and a corresponding algorithm to automatically convert scanned distance data to a machine toolpath. We enable on-tool modification of this toolpath in real-time through a texturizer module. Additionally, we contribute a despiralization algorithm that allows practitioners to parametrically modify the toolpath in desktop software (such as CoilCAM) if desired.

3.1 Scanner Mechanism

The CAS physical scanning hardware consists of a Keyence IL-300 triangulation-based laser displacement sensor (Figure 2 C) mounted on top of a vertically oriented elevator (made of aluminum U-channel) (Figure 2 D), which is raised and lowered above the deck of the DPW by a modified FUYU FSL40 stepper motor-driven linear actuator (Figure 2 E). The stationary structure of the linear actuator is mounted to the frame of the DPW, to the right (Figure 3 A), and slightly behind the wheel head. This architecture minimizes the intrusion of the scanner into the working area which facilitates using the DPW as a manual pottery wheel.

We identified three performance goals that guided our component selection. First, we sought to support the scanning of objects ranging in diameter from zero up to the diameter of the wheel head (350mm). Second, we targeted a scanning accuracy of 0.25mm and a resolution high enough to provide sufficiently smooth output data, which we estimated to be on the order of 0.1mm. Finally, we sought to avoid noticeable faceting of the output data while scanning at rates comparable to a typical linear printing speed of 50 mm/sec. These initial targets led us to select the Keyence IL-300 linear displacement sensor, which has a range of 290mm, a repeatability of 30um, and a basic sampling rate of 3 kHz. This sensor provides an analog voltage output, which we directly read using the onboard analog-to-digital converter (ADC) of a Teensy 4.1 microcontroller at an effective sampling rate (after averaging) of 1 kHz. At a linear print speed of 50mm/s, this gives a linear distance between points of 0.05mm. One challenge of using the Teensy onboard ADC was its limited effective resolution of 10 bits, which corresponds to a 0.3mm sensor resolution that exceeds our target. In practice we

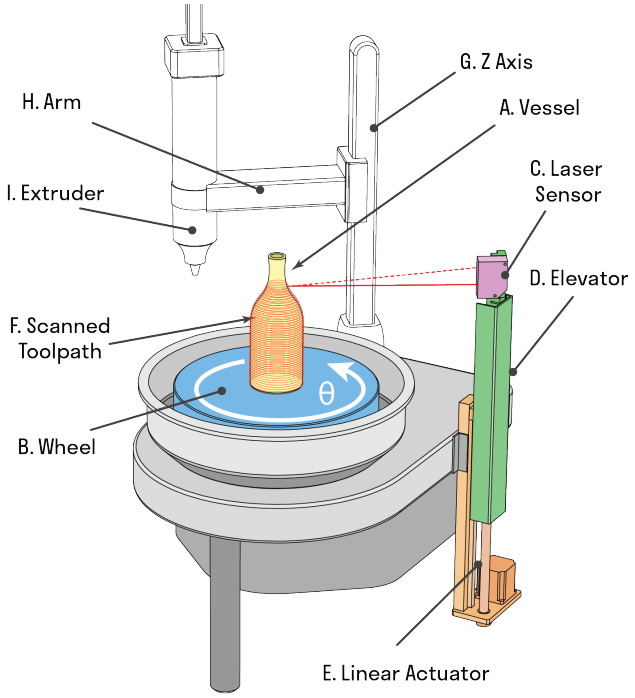

Figure 2: The CAS scanner mechanism consists of a triangulation-based laser displacement sensor (C) mounted to a vertical elevator (D) that is driven by a linear actuator (E). This placement avoids impinging on the working area of the Digital Pottery Wheel (DPW) to which the scanner is mounted. The practitioner positions the vessel to be scanned (A) at the center of the wheel (B). The CAS synthesizes a motion control signal to spin the wheel and simultaneously raises the laser sensor at a fixed distance for each revolution. This produces a spiral path (F) that captures the surface topography of the object. The resulting scan can then immediately be printed by the original 3D printer components of the DPW (H-G).

found our results were sufficiently smooth and accurate for our purposes. We see a future opportunity to take advantage of Teensy's high sampling rate to gain additional resolution by replacing our moving-window averaging algorithm with an oversampling and decimation approach [7].

The IL-300 uses a visible laser as the sensing mechanism. This provides the practitioner with visual feedback as to exactly where the scan is taking place, allowing for intuitive control over starting and stopping points. The IL-300 uses a 0.5mW Class 2 red laser diode, which is equivalent to a low-powered laser pointer. While the risk of eye damage is low, we aim the laser below eye level and obliquely relative to where a practitioner typically sits at the wheel.

3.2 Scanning Control System

The process of using the CAS starts with positioning an object to be scanned at the center of the DPW wheel head or throwing a vessel in place using the DPW in wheel mode. The latter has the

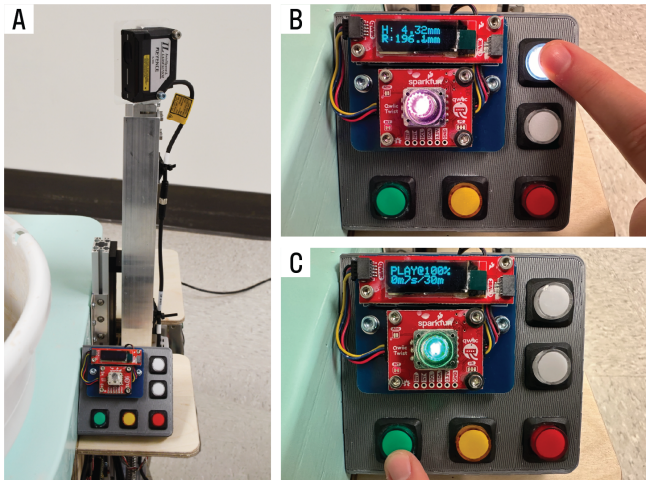

Figure 3: The CAS scanner and control interface. A. The assembled CAS mechanism and user interface are mounted to the right side of the DPW. B. The practitioner can adjust the scan start position by raising or lowering the elevator with the right-hand white buttons. C. When the scanner is at the desired height, the practitioner initializes the scanning recording process by pressing the red button. The practitioner can initialize scan playback by pressing the green button to print just the scan or the yellow button to first print a base, followed by the scan.

advantage of the object being de-facto-centered. Because the DPW uses a standard 14" wheelhead, we can also transfer thrown objects from a standard pottery wheel simply by moving the wooden build surface, or "bat," from the wheel to the DPW.

Next, the practitioner manually positions the laser sensor at the vertical starting point of the scan, using the "up" and "down" buttons on the CAS user interface (Figure 3 B). In some cases, such as when sampling just a portion of a vessel, it may be desirable to start somewhere other than at the surface of the wheelhead. Finally, the practitioner presses the "record" button, which starts the scanning process (Figure 3 C). Our scanning algorithm operates as follows: The CAS controller synthesizes a motion control signal, which feeds into the DPW control system to spin the wheel. Simultaneously, the CAS continuously raises the laser sensor a fixed distance (the scanning pitch) for each revolution of the wheel. The combined motion of the wheel and sensor causes the laser sensor to trace a spiral path on the surface of the object and results in a spiral sequence of data points that capture the surface topography.

To enable workflows where the practitioner goes directly from scanning to printing without any intermediate processing in the computer, we store a spiral toolpath as the scan result. This process is enabled by a close alignment between our spiral scanning approach, the spiral toolpaths used in clay 3D printing, and the polar coordinate system of the DPW. We further this alignment by matching the scanning pitch to the desired vertical "layer height" of the toolpath. Furthermore, rather than represent the toolpath as a series of absolute points, we represent the toolpath in the same native data format that is used by the DPW control system: a set of

motion streams – one for each axis of the DPW – comprised of a sequence of step and direction signals sampled at 100kHz. These streams can be directly played back into the DPW control system to reproduce the geometry of the scanned object during printing. Our approach enables infrastructural compatibility between the CAS and DPW and provides the conceptual simplicity of "playing back" the scanned recording.

The time-dependent nature of the recorded toolpath introduces a tight coupling between the scanning process and toolpath generation. This creates hidden complexities in the scanning algorithm because scanning does not occur at a constant angular speed – as might be possible if we were only concerned with capturing scan points – but rather occurs at a constant *surface speed* required for consistent printing. For example, a discontinuity in the radius of the scanned object is not simply recorded as another point but must be traversed over a period of time by the recorded toolpath as a radial-only move of the DPW arm. We address issues such as this by running a motion planning algorithm for a simulated DPW in parallel with the scanning control algorithm. When the scanner measures a point in polar (R-θ-Z) space, the simulated DPW is commanded to move to that point under the constraints of a constant linear printing velocity and a maximum wheel speed. If the laser measures a sudden step in the radius of the object, the simulated DPW will respond by moving radially to the new point at the print speed while pausing the simulated wheel. This typically occurs at the beginning of a scan because we assume the DPW starts at the wheel center, while the first point of a scan is at a non-zero radius (e.g., the radius of a cylindrical form being scanned). We seek to keep the scanning process relatively continuous and in step with the toolpath generation and recording process by a) scanning at the target linear print speed and b) buffering points from the scanner. As the buffer fills, we proportionally lower the rotational scanning speed of the wheel to keep scanning and recording approximately synchronized. During this entire process, it is the output of the motion planner – not the point data from the laser sensor – that is recorded to an SD card as multi-channel step and direction control streams. We record at a rate of 100 kHz and pre-allocate 30 minutes worth of storage (180MB) to make storage at these data rates possible with our chosen MCU. While we directly generate the motion of the DPW's simulated axes from the laser scan data, we synthesize the extruder control stream during the recording process based on path lengths and a tunable extrusion parameter. The laser sensor measures the exterior radius of the object, but the toolpath is recorded at the diametrical center of the extruder nozzle. We, therefore, assume a 6mm extrusion bead width to correspond with the DPW nozzle diameter and offset the laser data accordingly before feeding it into the motion generation algorithm.

The CAS controller hardware is constructed to seamlessly plug into the existing DPW modular control system and has outputs for each of the DPW axes (wheel, arm, z-axis, and extruder). Printing from a recorded scan starts when the practitioner presses the physical "play" button, which initiates a direct playback of the recorded motion streams into the DPW control system (Figure 3 C). For many pieces, it is advantageous to first print a clay base: this enables their use as watertight vessels and also aids with the adhesion of the clay walls to the wheel during printing. To facilitate this, we introduce an alternative playback approach triggered by a "print with base" physical button. In this mode, playback starts by silently reading through the recording, looking for when the wheel is first commanded to spin. Because the initial move is always radial – bringing the DPW arm from the center of the wheel to the radius of the scanned object – finding the moment when the wheel starts to spin also yields the starting radius of the scan. From here, the CAS synthesizes a toolpath that generates a multi-layer spiral base with a radius matching that of the beginning of the scan. Once the base is done printing, the standard playback routine is initiated to print the walls according to the scanned and recorded toolpath. All three operations described above – "scan", "play", and "play with base", can be stopped mid-stream by pressing their respective physical buttons a second time.

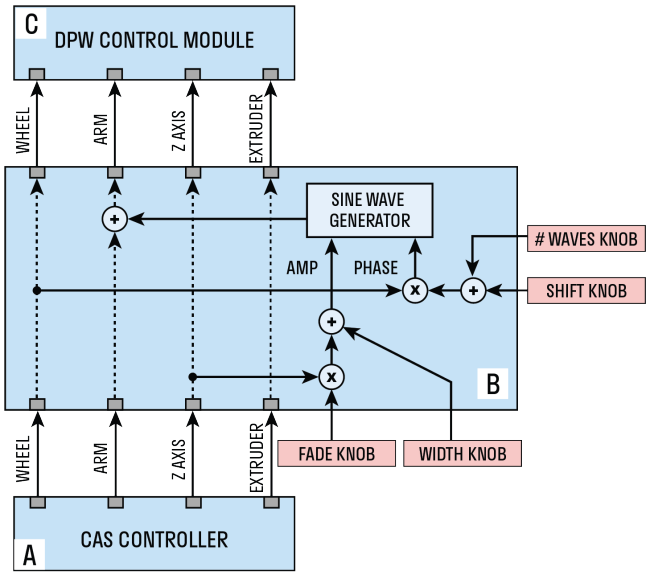

Figure 4: The texturizer module. When the texturizer module is inactive, the outputs from the CAS controller (A) are simply passed through to the DPW control module. When activated, the wheel and z-axis motion streams are fed into a texture generation algorithm (B), which synthesizes a sine wave signal based on these inputs. The sine wave signal is then additively overlaid onto the arm motion stream, creating a sinusoidal fluctuation of the radial position of the extruder nozzle while printing.

3.3 Texturizer Module

We developed a texturizer control module that enables a practitioner to introduce surface textures onto a scanned toolpath during playback and to control the generation of these textures in real-time. The texturizer module is inspired by CoilCAM's approach of using function generators to create toolpath variation [5]. We adapt this approach to a form that allows for the manipulation of real-time motion streams rather than pre-defined toolpaths and is compatible with the DPW modular control system.

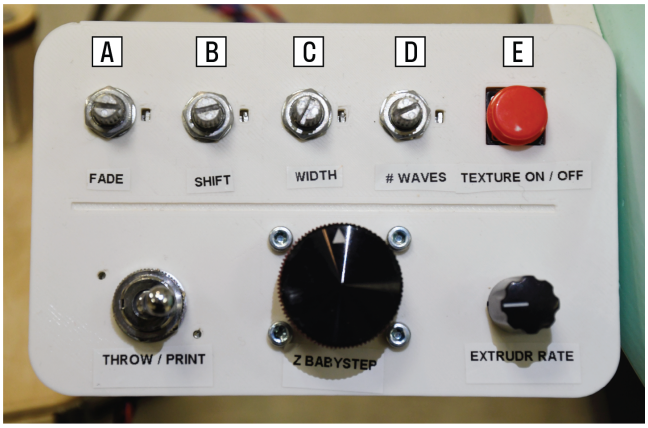

Figure 5: The texturizer user interface includes knobs to control (A) fade, (B) shift, (C) width, and (D) number of waves, as well as (E) a toggle switch to activate the texturizer. Note that we integrate it with the existing physical user interface controls of the DPW below the texturizer knobs

As shown in Figure 4, the texturizer module integrates between the CAS controller (Figure 4 A) and the DPW control system (Figure 4 C). When the texturizer is off, the CAS's playback outputs are simply passed through the texturizer module and into the DPW control system. With the texturizer enabled, these motion streams continue to pass through; however, the wheel and z-axis motion streams are fed into a texture generation algorithm (Figure 4 B), which synthesizes a sine wave signal based on these inputs and four physical control knobs. The texture signal is then additively overlaid onto the radial arm motion stream. The result is a sinusoidal fluctuation of the radial position of the extruder nozzle while printing, which manifests as a periodic surface texture. This process occurs in real-time. We created a texturizer user interface that enables control of the texturizer parameters through a bank of knobs (Figure 5). The UI also contains a toggle switch that activates and deactivates the texturizer functionality (Figure 5 E). The control knobs are read at a rate of 500Hz, and the sine wave generator loops at a rate of 10kHz. Because the texturizer operates directly on step and direction motion streams, which support step rates of up to 100kHz, a fast refresh rate for the generator algorithm ensures that the output motion streams remain smooth.

The sine wave generation algorithm is parameterized by the angular motion of the wheel, the position of the Z axis, and four physical knobs, which expose the following controls to the practitioner: "Wave Count", "Shift," "Width," and "Fade." The Wave Count knob (Figure 5 A) controls how many sinusoidal waves are generated per revolution of the wheel. The Shift knob (Figure 5 B) controls the offset between waves of adjacent layers. For example, a width of 3mm and an offset of 50% may produce a tight, symmetric woven pattern in the resulting surface. Mathematically, these offsets occur when a non-integer number of waves is specified; the remainder after a full wheel revolution appears as a shift between layers. To simplify this concept for the practitioner, we split the integer and fractional components of the sine wave frequency into wave count and shift, respectively. The Width knob (Figure 5 C) controls the width of each sine wave, as measured peak-to-peak. Finally, the Fade knob (Figure 5 D) affects the width of the sine wave as a parameter of the z position. This enables gradually fading surface textures. An unanticipated result is that surface texture can appear to invert with negative fade values when the modified width of the sine-wave first shrinks and then grows in negative value. These four controls affect the parameters of a sine equation that is evaluated every 100us.

$$\Delta R = A * sin(\phi)$$

where:

ΔR is the output of the generator, which is added to the radial motion stream of the arm each cycle.

A is the sine amplitude and is the sum of the width knob Kw and a fader component Fc:

$$A = Kw + Fc$$

The fade component Fc is the running sum of the Fade knob Kf multiplied by the differential motion of the Z axis since the last evaluation of the sine equation:

$$Fc = \Sigma(Kf * \Delta Z)$$

ϕ is the phase angle of the sine wave and is the running sum of the total wave frequency (wave count knob Kc + shift knob Ks) multiplied by the differential motion of the wheel since the last evaluation of the sine equation:

$$\phi = \Sigma((Kc + Ks) * \Delta W)$$

By using running sums of difference equations, the transition from one set of control parameter values to another occurs smoothly.

The texturizer's output is a combined function of the texturizer settings, the machine printing parameters (e.g., nozzle diameter, layer height, extrusion rate), and the clay's material properties (moisture content, clay body type). As is the case with clay 3D printing in general, these properties collectively impact the plasticity of the extruded coil and the stability of the printed form. For example, a high-frequency oscillation of the texturizer may break a thick, low-moisture sculpture-body coil but work fine for a thin, well-hydrated throwing body coil. The texturizer enables the craftsperson to rapidly alter oscillation parameters on the fly based on the observed material output (Figure 6), and the craftsperson can exercise skill and discretion in tuning the texturizer to achieve their desired effect.

3.4 Despiralization Algorithm

As previously stated, the CAS stores a time-parameterized recording of a spiral toolpath that reproduces the surface of the scanned object when printed. While this approach enables a primary workflow of directly printing from a recorded scan, we also sought to explore workflows that involve modifying the toolpath using desktop CAM tools such as CoilCAM [5], which operate on sequences of closed paths at equally spaced Z heights. To support this, we developed a Python script that converts the step and direction-encoded scanned toolpath into a spiral sequence of points in 3D

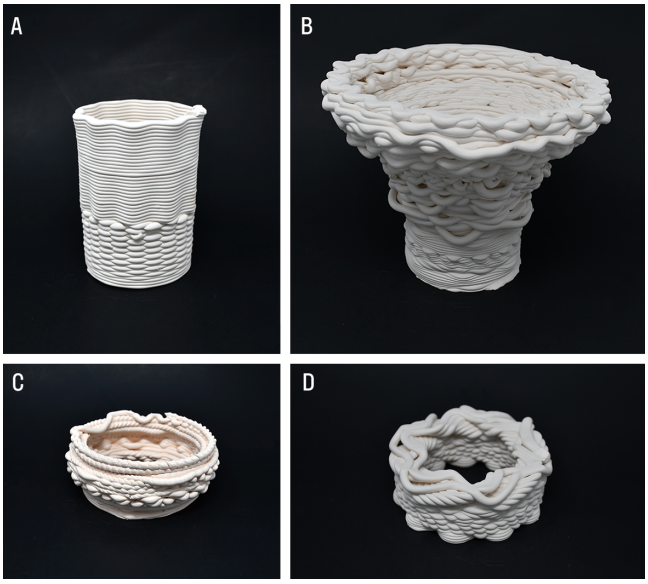

Figure 6: Pieces produced through the rapid and irregular tuning of texturizer parameters.

polar space. We then execute a *despiralization* algorithm that converts this spiral into a sequence of closed paths at fixed layer heights. We establish horizontal planes at Z positions corresponding to the scanning/printing pitch and then project points from the spiral onto the nearest downward plane. The process is made easier by retaining the polar coordinate system, as each loop consists of projected points from a single revolution of the wheel. One challenge of this approach is that loops do not always "close up", because the projected points are originally coming from different Z heights, and the scanned object may have a non-vertical slope. To account for this, we simply take the difference in radius between the first and last point in the loop and then distribute this difference evenly across each point so that the start and end points share the same radius. The final step in the process is to convert from polar to Cartesian coordinate space.

4 LIMITATIONS

The CAS scanning technique is limited to capturing 3D geometry with either a closed top or an open planar top. Our technique is also not suitable for geometry with holes– e.g. we cannot produce an accurate scan of a cup with a handle. These limitations are acceptable because they directly align with the affordances of both wheel throwing and clay 3D printing. In wheel throwing, artists produce continuous cylindrical forms with a single planar opening. Clay 3D prints are also frequently continuous and lack gaps and substantial overhangs because many clay extruders cannot create a clean retract, and all cannot print with support material.

We discovered that continuously observing the CAS system while printing with the texturizer produces a slight disorienting effect. This is because the operator is watching a spinning object and the oscillating extruder simultaneously. The effect is mild and dissipates quickly. We believe we could remediate it by placing a mirror behind the extruder so that the operator can observe the profile of the vessel rather than the wheel and extruder simultaneously. This strategy is well established in traditional wheel throwing.

In our validation of CAS, we collected verbal reflections from the expert potter we commissioned to use our system. We refrained from conducting formal thematic analysis on this secondary data, which could have yielded additional insights. Our primary objective was to assess the quality of artifacts feasible to produce with CAS. We see opportunities for a future qualitative study evaluating practitioner experience across multiple participants.

5 VALIDATING CAS WORKFLOWS WITH PROFESSIONAL MANUAL ARTIFACTS

We sought to understand how CAS enables the creation of functional and aesthetic pottery through the integration of manual material production and 3D printing. To validate the performance and versatility of the CAS as a pottery production tool, we developed four workflows for using material engagement to design 3D printed vessels. We commissioned a professional potter to produce manually thrown vessels as input for these workflows. *Isaih Porter*[1] is a production potter with ten years of experience who makes the majority of his income from making and selling functional ceramics. We conducted a four-hour design session where Isaih threw four different vessels of his choice with a maximum height of 12 inches and a maximum diameter of 8 inches (Figure 7). After Isaih threw each vessel, we scanned it and walked him through applying one of the workflows to create a new vessel. We kept all scans of the thrown vessels and used them for additional workflow tests and artifact production following the conclusion of the session. We compensated Isaih $400 USD for his time and labor.

As a result of this study, we generated eight unique CAS artifacts produced solely by modifying scans of four professional manually thrown vessels. We also created two additional CAS artifacts from scans of pre-existing noncylindrical and cylindrical vessels. We assessed the quality of these vessels by their function as pottery artifacts–including water-tightness, printability, and structural integrity; accuracy in comparison to the manual originals; and stylistic variety. The CAS-produced vessels derived from scans of Isaih's work constitute our primary results because they demonstrate the capacity of the CAS system to reproduce and modify manually produced ceramics. We collected secondary data through audio and video recordings of Isaih's use of CAS and his reflections using the technology. We selected representative samples from this secondary data to illustrate the construction process of the CAS artifacts. In the remainder of this section, we describe each workflow in detail using one or more artifacts produced through a combination of manual input and CAS features as illustrative examples.

5.1 Reproduction of Thrown Vessels

We explored CAS as a mechanism to reproduce manually thrown vessels in clay 3D-printed form. This *reproduction workflow* begins with scanning an object using the CAS system. Before starting the scan, we toggle the DPW into throwing mode using the switch on the bank of controls located on the left side of the wheel head.

[1] https://www.isaih.xyz

Figure 7: Vessels thrown by Isaih on the DPW that we used as input for the CAS workflows. A) Mug. B) Bottle. C) Vase. D) Spout.

This moves the extruder out of the way, enabling an easier scan. Once the scan is complete, to print the scanned path, we toggle the switch back to printing mode. We adjust the extruder nozzle to be 1 mm above the wheel head and start the print on the CAS module in base mode. When printing is complete, the artifact is ready for additional post-processing or drying operations.

We validated the reproduction workflow by scanning and printing three vessels thrown by Isaih: a 14.5 cm tall vase with a 13.5 cm maximum diameter (Figure 7 C), a 14.5 cm tall mug with three horizontal indentations and 9.4 cm maximum diameter (Figure 7 A), and a 20.5 cm tall bottle with 7.7 cm maximum diameter (Figure 7 B). In each instance, we were successfully able to use the CAS to print the scan of the thrown vessel. The height of each printed vessel closely corresponded with the height of the original thrown forms. The outer diameter of each vessel was slightly wider than the original thrown form. This was due to the layer compression that takes place during clay 3D printing. We calculate the printing path to correspond to a 6 mm nozzle diameter and, thus, a 6 mm bead width. However, in practice, we found that this produced a 7.5 mm-wide bead during printing when paired with our layer height of 2.5mm. This additional 1.5 mm accounted for the increase in diameter we observed in each printed form and could be compensated for in a future iteration of the CAS scanning algorithm.

Isaih observed that aside from the increase in diameter, most forms exhibited minimal differentiation. The exception was the vase. He observed a slight flattening of the curve of the vessel in the lower section. We believe this does not reflect a distortion in the scan because it is not present in the narrower vessels. Rather, this quality was likely the result of a slight collapse of the form during printing- a factor that is common in clay 3D printing forms with sharper overhangs. We could address this by adapting the toolpath to maintain a consistent layer thickness as demonstrated by Friedman-Gerlicz et al. [14]

We also experimented with manually modifying a version of the vase on the wheel. After concluding the design session, we reprinted a version of the vase, and Jennifer used the DPW in throwing mode to smooth the printed coils with a sponge and a rib on the lower portion of the vessel (Figure 13 C). This resulted in a surface quality that mimicked the original thrown vase (Figure 13 D). The process further distorted the profile of the vessel from the original scan.

5.2 Reproduction of Asymmetrical Forms

Wheel throwing results in cylindrical radially symmetric forms, whereas clay 3D printing supports the production of asymmetric forms. The CAS system supports scanning and printing asymmetric forms directly on the wheel. Figure 12 shows the results of scanning and printing an asymmetric 3D-printed vessel by the artist Jeff Suina [34] (with added surface texture post-scan). Potters create only cylindrical forms while the wheel is spinning, but they frequently manually alter vessels after throwing to create partially asymmetric shapes– for example, creating indentations in a cup for easier holding or bending the lip of a vessel to produce a pitcher. We used the combined affordances of CAS scanning and clay 3D printing to capture and reproduce asymmetric forms to develop a workflow for integrating wheel throwing and asymmetrical design.

The *asymmetrical reproduction workflow* begins with throwing a vessel on the wheel, followed by making manual adjustments to render a desired portion of a wheel-thrown vessel asymmetric. Following this, we adjust the CAS scanner starting position to the lower limit of the asymmetric portion and scan to the upper limit of the portion. We then throw a new cylindrical vessel on the wheel and print the asymmetric portion on top. The order of these last two operations can be reversed depending on the skill of the potter– e.g., a skilled potter can print an asymmetric lower portion and then center a ring of clay on top and throw a cylindrical vessel from that. Furthermore, because CAS records scan data in relative steps and directions rather than absolute points, we can use the existing DPW physical user interface to adjust the diameter of any printed scan on the fly. As a result, the potter can throw a vessel of arbitrary diameter and adjust the dimensions of the asymmetrical scanned portion to ensure it fits without measurement and modification in desktop software.

Figure 8: Workflow of scanning and printing pieces thrown on the wheel. A) Isaih throwing a mug on the wheel by hand. B) Thrown mug being 3D printed on the wheel from a scan of the original mug. C) Thrown mug. D) Printed mug. E) Isaih throwing a vase on the wheel by hand. B) Thrown vase being 3D printed on the wheel from a scan of the original vase. C) Thrown vase. D) Printed vase.

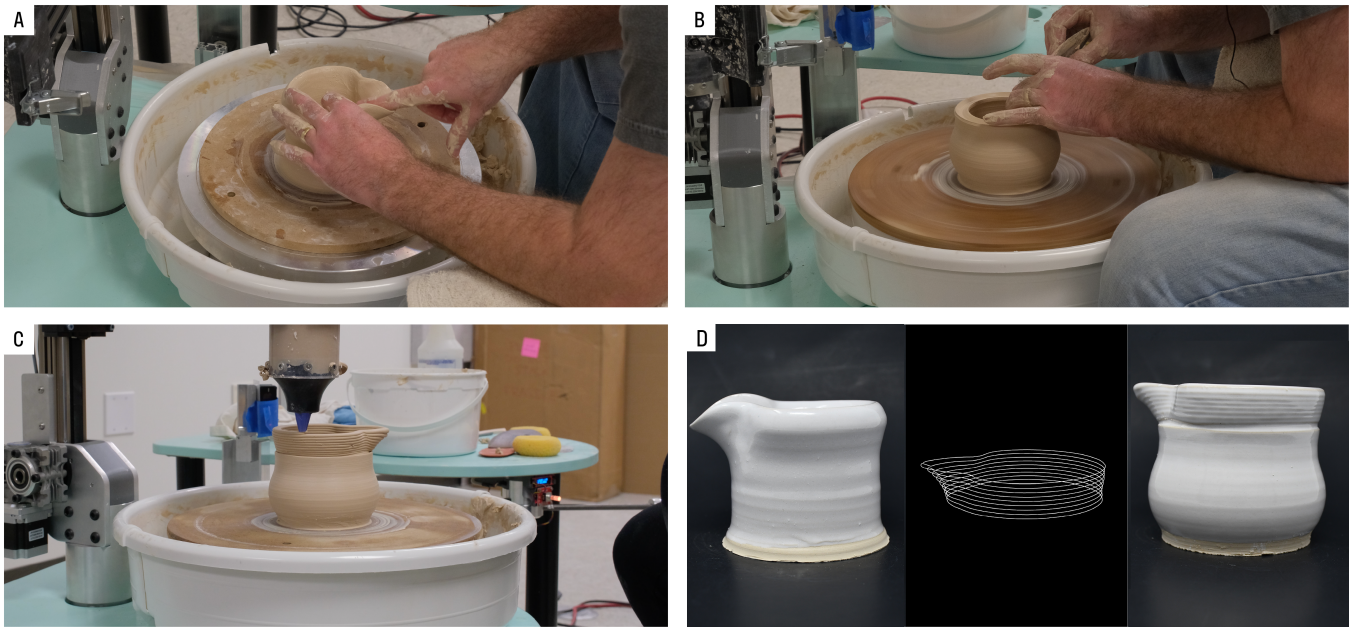

Figure 9: Reproduction of asymmetrical forms. A) Isaih forming a spout manually on his thrown vessel. B) Isaih throwing a second vase with a cavity on the top of the lip to support the first 3D-printed layers. C) 3D-printing the scanned spout from A. D) Comparison between the thrown spout, the scanned bottom part of the thrown spout, and the resulting pitcher made of a thrown body and a 3D-printed spout.

We validated the asymmetrical reproduction workflow by requesting that Isaih throw a short cylinder and manually shape it into a spout (Figure 9 A). We then scanned the spout. We stopped the spout scan at the point where we could still maintain a planar toolpath, meaning we could not capture the upper 2-3mm (Figure 9 D-middle). Following this, Isaih threw a pitcher base of arbitrary diameter with a groove along the top edge to catch the 3D printed coil (Figure 9 B). We then printed the scanned portion on top of the thrown vase (Figure 9 C) to produce a pitcher with a spout resembling the Isaih's manually created design (Figure 9 D-right).

5.3 Integration with Desktop CAM

In addition to on-tool reproductions, we developed an asynchronous CAM-based design workflow for the CAS where the practioner uses an external desktop software to apply modifications to the scan before printing it on the DPW. This approach enables the digital modification of scanned vessels to include complex surface textures and non-cylindrical geometry. To demonstrate this workflow, we used CoilCAM, a parametric system for clay 3D printing that enables the specification of mathematically defined machine toolpaths [5]. CoilCAM is a node-based programming system built as a plug-in to the Grasshopper software. It is developed around the unit toolpath generator node, a cylindrical toolpath that can be modified in radius, scale, gradient, rotation, and translation using function and Boolean operators. The CAS scanning output, paired with our despiralization algorithm, enables CAS scans to be modified through CoilCAM.

To begin the desktop CAM workflow, we retrieve the scanned path from the micro SD card located on the CAS module and download the binary scan files onto a desktop computer. Using our despiralization algorithm, we generate a text file containing a list of points with a constant z-height in each layer. To make this list compatible with CoilCAM, we use an additional custom Python script in Grasshopper that takes the text file as input and returns the number of layers, the average radius of each layer, the number of points per layer, and the layer height. We use these parameters as input to the CoilCAM toolpath unit generator node. This enables the creation of a radially symmetrical toolpath that we can further modify using CoilCAM native functions and Boolean operators. Once we complete editing the toolpath, we save the toolpath data generated by CoilCAM in a G-code file. We then upload and print the G-code on the DPW.

We validated the CAM integration workflow by applying two interwoven square wave textures to a small hand-thrown bottle created by Samuelle (Figure 10 A). We scanned the bottle with the CAS and converted the scan in Grasshopper as described above (Figure 10 B). We then used a combination of CoilCAM shaping operators to create the texture on the bottle (Figure 10 C). We used a square wave function on the radius of the toolpath unit generator to create 11 bumps per layer and another square wave on the gradient profile parameter to make the texture visible on the body of the bottle but not on the lip. We added a sinusoidal function on the rotation parameter of the toolpath unit generator to create a vertical oscillation in the texture. We duplicated this node structure to generate a second identical instance of the toolpath unit generator but offset its sinusoidal rotation by 180°. Finally, we used a Boolean

union operator to combine the two instances, the offset in rotation creating an interference pattern between the two textures. We found the inside texture more aesthetically pleasing and decided to invert the final texture with an inside-out operation. The generated toolpath created with CoilCAM is depicted in Figure 10 D and the resulting print in Figure 10 E.

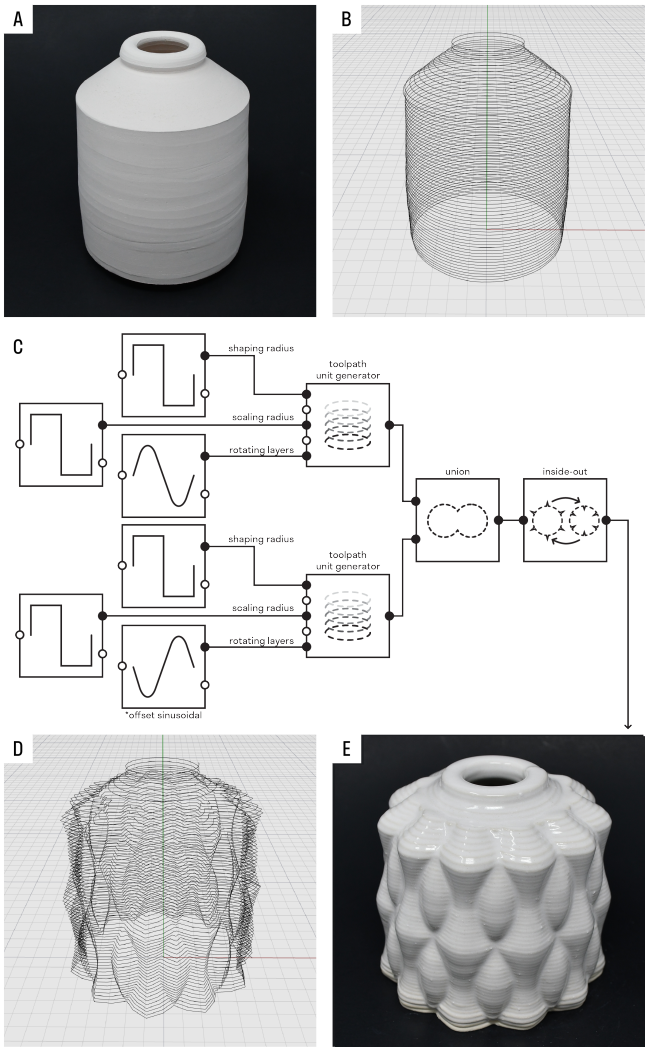

Figure 10: Preliminary example of using desktop CAM to modify a scanned path. A) A bottle hand-thrown by Samuelle. B) The initial scanned path represented with CoilCAM-compatible parameters. C) The CoilCAM program used to modify the scanned path. We used a Boolean union between two toolpath unit generators with a square wave radial texture offset in rotation by 180°. D) The toolpath generated by CoilCAM. E) The resulting print.

We tested a similar workflow on two scans of Isaih's vessels– the mug (Figure 7 A) and the vase (Figure 7 C) to create a mug with concave handles and a vase with a bumpy texture. To create the modified mug, we applied two Boolean difference operators to subtract two instances of the mug offset by 70 mm on each side of a third instance of the mug (Figure 11A). As shown in Figure 11B, this created two cavities that follow the curvatures of the mug, which can be used as handles. To create the modified vase, we first multiplied together a square wave and a sinusoidal function on the radius parameter of the vase to produce discrete but soft-looking bumps. We added a sinusoidal function to the gradient profile to make the bumps gradually fade in and out of the surface and a sinusoidal function on the rotation to angle them. We duplicated this node structure and changed the bump frequency on the radius and the gradient profile of this new instance. We combined the two resulting toolpath unit generators with a Boolean union operator to create variation in the bump pattern. We show the CoilCAM toolpath and the completed print in Figure 11 C and D.

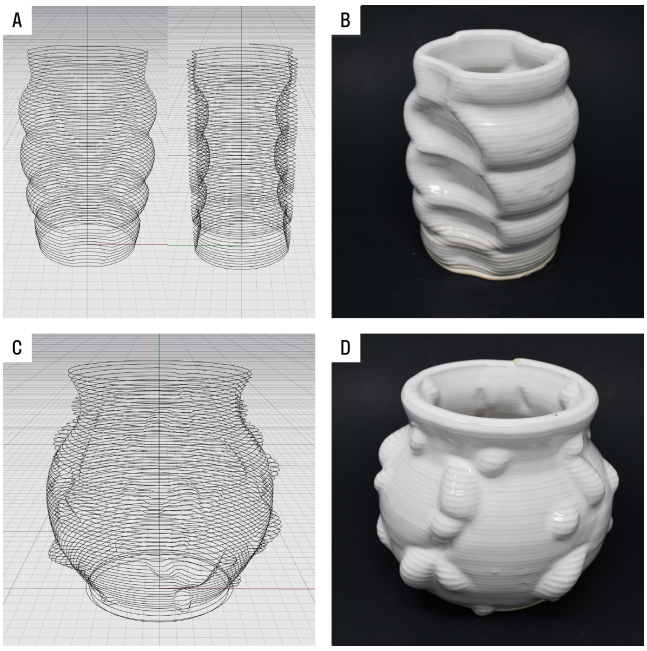

Figure 11: Scanned vessels thrown by Isaih and modified using CoilCAM. A) The toolpath generated by CoilCAM of the mug with Boolean differences of the same toolpath on both sides of the mug to create concave handles. B) The resulting print. C) The toolpath generated by CoilCAM of the vase with bumps. D) The resulting print.

5.4 On-Tool Modification of Scanned Form With Texturizer

The CAM integration workflow provides access to the rich textural affordances of clay 3D printing; however, it necessitates that the practitioner have a degree of CAM software expertise. The CAS texturizer module allows a practitioner to augment scanned forms with surface texturization through a workflow that is entirely located on the wheel. Our *on-tool texturization workflow* comprises a spectrum of approaches for wheel-based real-time design modification that range from pre-planned automated operation to highly improvisational live tuning.

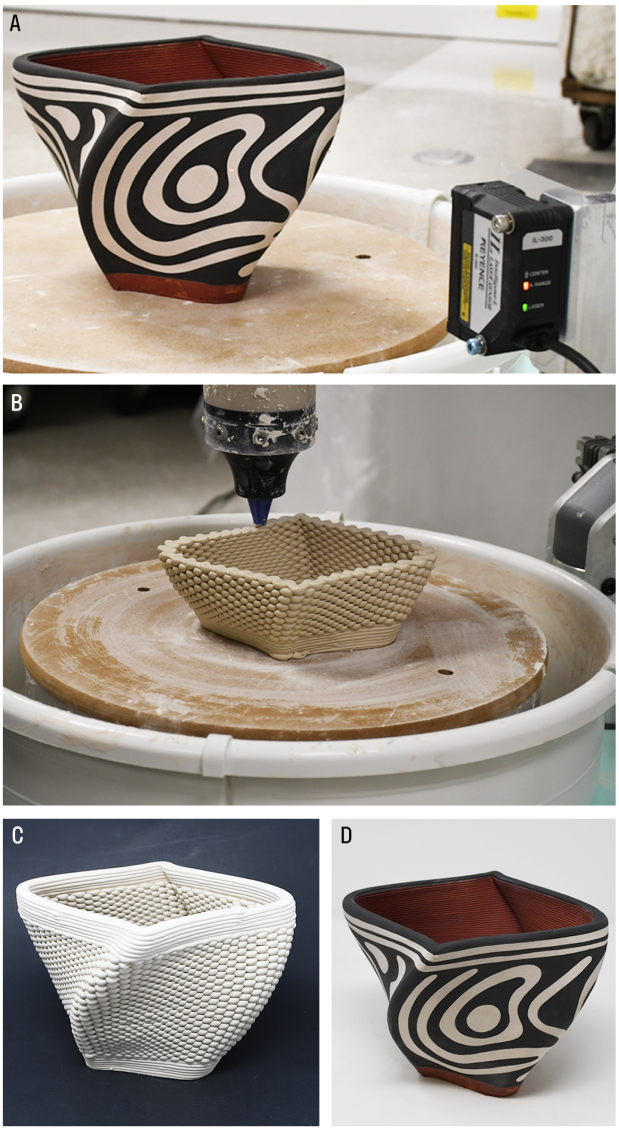

Figure 12: Jeff Suina's 3D printed vessel. A) Vessel, originally designed and 3D printed in clay by Jeff Suina, is scanned by the CAS system. B) Vessel is reprinted from scan with surface texturing incorporated live through the texturized module. C) Scanned and texturized vessel. D) Original vessel.

First, we scan the desired vessel. Following this, we start the print and introduce oscillation into the extruder by manipulating the controls on the texturizer control panel. In a pre-planned workflow, we begin with the texturization set to off and adjust the knobs to our desired parameters. We then start the print, and at the desired moment, we initialize the texturization by pressing the texture on/off button (Figure 5 E). Figure 12 shows the result of this process where we toggle a preset texture on after approximately eight layers of the scan of Jeff Suina's vessel. The texture parameters in this example are as follows: # of Waves: 40, Width: 4 mm, Shift: 35 %, Fade: 0, resulting in a weave-like structure.

In an improvisational workflow, we start with all texturizer parameters at zero and activate the texturization at the start of the print. We then gradually adjust the parameters while the scan is printed to achieve variations in the form as the print progresses. Because the texturizer module directly modifies the step and direction signals of the original scan, changes in parameters are immediately visible in the printed output. This allows the operator to quickly judge the effect of an adjustment and make further modifications. Figure 13 shows the results of live tuning on a scan of Isaih's vase (Figure 13 A). We toggle on the texturizer halfway through the print and gradually ramp up the Width parameter to show visible waves (Figure 13 B). We then gradually increase the Shift value to produce an offset. When the print nears the top, we ramp down both the Shift and Waves to zero to ease out the waves and print a clean rim.

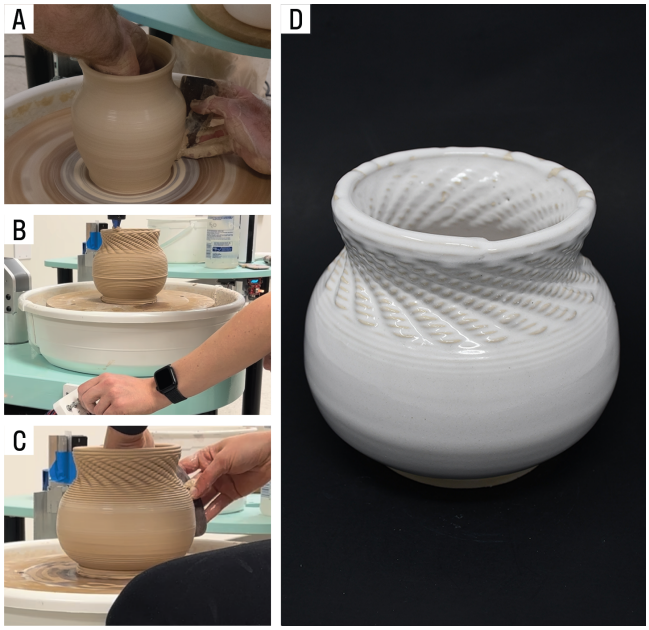

Figure 13: Demonstration of multi-stage throwing, texturization, and manual-modification workflow. A. Isaih throwing the vase on the DPW. B. Jennifer printing the scan of the vase while updating the texturizer parameters in real-time to produce a texturized upper portion that fades in and out. C. Jennifer manually smoothing the printed coils of the vessels using the DPW throwing mode. D. The finished vessel.

We validated the on-tool texturization workflow by first requesting that Isaih use the texturizer module to create a modified version of his thrown bottle (Figure 1). During this process, Isaih tested different parameters, producing both desirable and unexpected results (Figure 1 C). At the midpoint of the bottle, he experimented with the fade parameter with a wave parameter value of seven and a width parameter of 20-30 mm. By setting a negative fade value, we observed how the sine wave gradually inverted, producing an alternating set of interspersed large bumps every 12-13 layers. This

pattern appealed to both the authors and Isaih. Following the study, Jennifer created a new print of the bottle that replicated this pattern across the entire body of the vessel (Figure 1 D). She began with a Fade, Width, and Shift value of zero and a Wave Count value of seven. She then gradually increased the width value to 40 over the course of 14 layers. Then, she decreased the Fade value to -4 mm/cm. She waited until the DPW printed a fully inverted set of waveforms, then increased the fade value to 4 mm/cm. She repeated this process until the print reached the neck of the bottle, at which point she set the Fade back to zero and gradually tuned the Width to zero as well. This process took two attempts to perform satisfactorily. Once the finished bottle was leather hard, Jennifer trimmed the base and top in a chuck on the wheel to produce a smooth texture comparable to the original thrown form. Figure 1 E shows a comparison of the original thrown bottle, an untextured replication, and the final texturized version created by Jennifer.

6 DISCUSSION

We discuss the CAS workflows through two lenses: the benefits and potentially negative consequences of reducing friction in craft reproduction and the role of the texturizer in introducing risk and play into the digital fabrication process.

6.1 Benefits and Consequences of Reducing Friction in Skilled Craft Reproduction

In our validation, we demonstrate three workflows that allow for reproductions of existing forms with varying degrees of fidelity to the original. The throwing reproduction workflow demonstrates how CAS can rapidly and accurately reproduce manually thrown vessels in 3D printed forms. While thrown and 3D printed ceramics have different aesthetic and functional qualities, the integration of CAS with the DPW throwing capabilities demonstrates that it is possible to immediately modify a scanned and printed form to resemble the qualities of a thrown form on the same tool. This can be performed by an untrained practitioner (e.g., Jennifer) with acceptable results or a skilled potter with highly refined results [25]. Overall, this process suggests a new direction for production pottery, in which portions of the process for reproducing a vessel might incorporate 3D printing without requiring desktop software intervention.

The asymmetrical reproduction workflow demonstrates how the CAS enables practitioners to design irregular forms for 3D printing by leveraging existing on-tool manual sculpting skills. This approach further extends the previous reproduction workflow into the domain of asymmetric vessels. As Isaih's spout example indicates, the CAS is limited in the range of asymmetric geometry it can fully capture. We note that manual manipulation of the printed asymmetric scan offers an opportunity to overcome this limitation because skilled practitioners can manually smooth and re-adjust the printed portion if desired. The asymmetrical workflow also suggests opportunities for creating variations. For example, the practitioner could manually sculpt a key design feature, like a set of knob handles, and then use the combination of the CAS and the DPW to reproduce the asymmetric feature on vessels of different diameters and heights based on application or aesthetic preference.

Integrating the CAS with CoilCAM shows how our domain-specific scanning representation can significantly reduce the friction in transitioning between manual clay work and digital CAM-based design. In these previous examples, the general form of the 3D-printed vessels is defined manually, while the added textures are defined computationally through combinations of mathematical operations. By representing the scanned vessel as a continuous toolpath, the CAS facilitates this rapid transition between the two modes of production. This representation aligns with how a typical slicer software generates spiralized toolpaths for 3D-printing applications, enabling easy scan manipulation with CAM tools. The CAM integration workflow suggests a future direction where CNC and software design technologies can harness shared representations to support bi-directional design and fabrication workflows.

All of these workflows are incapable of fully reproducing manually constructed forms at present. This is partly due to the limitations of our scanning technique and the limitation of clay 3D printing compared to manual throwing and hand manipulation. Yet, while CAS cannot entirely reproduce manual work, it substantially lowers the barrier to reproducing 3D artwork geometry, *regardless of original authorship*. While reducing friction in reproduction can lower barriers for learners and experts, such tools must also be weighed against how they may reshape or destabilize professional craft. With the present implementation of CAS, the greatest risks for unlicensed reproduction exist when scanning and re-printing other 3D-printed vessels because the system can accurately reproduce 3D printed geometry with high fidelity as we demonstrate with Jeff Suina's work (figure 12). Intellectual property conflicts in craft production existed well before the integration of digital fabrication, but emerging technologies like the CAS could exacerbate these tensions. We see opportunities to evaluate how artists perceive the risks and benefits of tools like CAS through further dialog and collaborative design efforts. For the present moment, we have attempted to pursue the development of this technology in a value-sensitive and responsible manner. First, we received permission from the artists whose work we scanned. Second, we positioned the scanner as a subtool within a system to extend skilled manual production rather than circumvent it.

6.2 Enabling Risk and Play in Real-Time Digital Fabrication Adjustment

The on-tool texturizer workflow demonstrates several opportunities for extending manual throwing practice. Using constant settings, a practitioner can easily add surface features that are impossible through traditional wheel throwing and would otherwise require learning CAM or delay transitioning from desktop software to fabrication. We argue that the texturizer's real-time parameter tuning opens new dimensions for integrating craft and digital fabrication. First, in the context of CAS, the texturizer can augment manually crafted form with digitally driven variation in a material-centric context. Second, as an integration into the DPW, the texturizer further demonstrates a technical approach to how we might engineer digital fabrication to support real-time dynamic control systems. Third, the CAS texturizer is a powerful addition to developing improved digital fabrication methods for cementitious materials (e.g., clay, concrete, and biomaterials) because it supports precise

and digitally modulated real-time adjustment of pre-planned toolpaths, thereby enabling active adjustments in response to dynamic material and structural behaviors.

These opportunities represent two ways to reconceptualize CNC operations away from a fully automated technology paradigm. The first direction is CNC operation as an extension of *skilled craftsmanship*. By this, we mean tools that support a quality of outcome directly dependent on the operator's manual skill and coordination. This framing aligns with Pye's conceptualization of craft as the workmanship of risk [28]. Like any skilled craft, the texturizer requires practice. Isaih recognized this quality, stating that he found the real-time nature of the texturizer compelling and that he would like to practice with it further. Moreover, regardless of experience, tuning the texturizer introduces new forms of risk. An operator unfamiliar with the tool or the material constraints of clay may produce a work that does not align with their design intent. Alternatively, an operator with manual dexterity, timing, and material sensitivity may produce results that push the expressive boundaries of clay 3D printing as a medium.

The texturizer offers a second way of reconceptualizing CNC production through the lens of *playful experimentation*. The original DPW control system was inspired by reconfigurable tools in music production– namely modular synthesizers. The texturizer further extends this model by enabling dynamic adjustment of control signals with immediate material feedback. This process, paired with the plasticity of clay 3D printing, creates an interaction where the operator can experimentally react to the machine and the material and, in the process, generate new design ideas and make discoveries about material behavior. This interaction aligns with Andersen et al.'s concept of digital crafts-machine-ship, where digital fabrication involves watching the machine and responding to its actions, creating a space for conceptualizing ideas about making and envisioning alternatives [2].

We see opportunities for the CAS to scaffold playful experimentation in both material and software design contexts in ways that are not feasible with conventional CAD-CAM-CNC workflows. For material-based design, the ability to scan and repeatedly re-print the same form provides craftspeople with a common starting point to manually explore variations on the wheel– either with the texturizer or by hand. This process could encourage experimentation by reducing the labor required to re-create the base form. For software-based design, the CAS could scaffold digital experimentation by providing a low-friction means of creating a base form manually, in a medium of existing fluency. Unlike existing workflows that require designing digital forms de novo in software, this approach could ensure that experienced potters– who may not be experienced digital designers– have a springboard from which to achieve desirable variations through playful, incremental experimentation in software while softening the digital workflow learning curve.

7 CONCLUSION

3D printing, like many contemporary modes of digital fabrication, is primarily driven by software design. By integrating a novel domain-specific scanning technique with real-time dynamic printing variation, we show that it is feasible to design for 3D printing by starting in, and if desired, by remaining exclusively in the craft material domain. The CAS system offers a novel set of interactions to expand the expressiveness and workflows available within clay 3D printing. Furthermore, as digital fabrication technologies encompass a greater variety of dynamic and non-heterogenous materials, our research suggests one pathway for informing digital fabrication and CNC operation through material and machine behavior, thereby expanding the range of what we can feasibly fabricate through manual-digital workflows. While clay 3D printing is unavoidably (and often delightfully) messy, we show how to substantially improve the process without making a mesh.

ACKNOWLEDGMENTS

We would like to thank Timea Tihanyi, Lynda Weinmann, Joey Watson, and Linda Haggerty for providing feedback on early versions of CAS. We also thank Isaih Porter for participating in our research. We extend special thanks to Jeff Suina for giving us permission to test the system with his artwork. We appreciate the assistance of fellow Expressive Computation Lab members at UCSB. The evaluation components of this research were funded in part by the NSF IIS Future of Work at the Human-Technology Frontier Program (Award: 2026286) and the NSF IIS Human-Centered Computing Program (Award: 2007094).

REFERENCES

[1] Artec 3D. 2024. Artec Studio. https://www.artec3d.com/3d-software/artec-studio
[2] Kristina Andersen, Ron Wakkary, Laura Devendorf, and Alex McLean. 2019. Digital crafts-machine-ship: creative collaborations with machines. *Interactions* 27, 1 (dec 2019), 30–35. https://doi.org/10.1145/3373644
[3] Marco Attene, Marcel Campen, and Leif Kobbelt. 2013. Polygon mesh repairing: An application perspective. *Comput. Surveys* 45, 2 (March 2013), 15:1–15:33. https://doi.org/10.1145/2431211.2431214
[4] Javad Baqersad, Peyman Poozesh, Christopher Niezrecki, and Peter Avitabile. 2017. Photogrammetry and optical methods in structural dynamics – A review. *Mechanical Systems and Signal Processing* 86 (March 2017), 17–34. https://doi.org/10.1016/j.ymssp.2016.02.011
[5] Samuelle Bourgault, Pilar Wiley, Avi Farber, and Jennifer Jacobs. 2023. CoilCAM: Enabling Parametric Design for Clay 3D Printing Through an Action-Oriented Toolpath Programming System. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23)*. Association for Computing Machinery, New York, NY, USA, 1–16. https://doi.org/10.1145/3544548.3580745
[6] Paolo Cignoni, Marco Callieri, Massimiliano Corsini, Matteo Dellepiane, Fabio Ganovelli, and Guido Ranzuglia. 2008. MeshLab: an Open-Source Mesh Processing Tool. In *Eurographics Italian Chapter Conference*. 8 pages. https://doi.org/10.2312/LOCALCHAPTEREVENTS/ITALCHAP/ITALIANCHAPCONF2008/129-136
[7] Atmel Corporation. 2005. AVR121: Enhancing ADC resolution by oversampling. (2005).
[8] Audrey Desjardins and Timea Tihanyi. 2019. ListeningCups: A Case of Data Tactility and Data Stories. In *Proceedings of the 2019 on Designing Interactive Systems Conference (DIS '19)*. Association for Computing Machinery, New York, NY, USA, 147–160. https://doi.org/10.1145/3322276.3323694
[9] Sean Follmer, David Carr, Emily Lovell, and Hiroshi Ishii. 2010. CopyCAD: remixing physical objects with copy and paste from the real world. In *Adjunct proceedings of the 23nd annual ACM symposium on User interface software and technology (UIST '10)*. Association for Computing Machinery, New York, NY, USA, 381–382. https://doi.org/10.1145/1866218.1866230
[10] Sean Follmer and Hiroshi Ishii. 2012. KidCAD: digitally remixing toys through tangible tools. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. Association for Computing Machinery, New York, NY, USA, 2401–2410. https://doi.org/10.1145/2207676.2208403
[11] Sean Follmer, Micah Johnson, Edward Adelson, and Hiroshi Ishii. 2011. deForm: an interactive malleable surface for capturing 2.5D arbitrary objects, tools and touch. In *Proceedings of the 24th annual ACM symposium on User interface software and technology (UIST '11)*. Association for Computing Machinery, New York, NY, USA, 527–536. https://doi.org/10.1145/2047196.2047265
[12] formlabs. 2022. *3D Scanning and 3D Printing for Reverse Engineering and Other Applications*. White Paper. https://3d.formlabs.com/3d-scanning-for-reverse-engineering-restoration-metrology/#form

[13] J.G.D.M. Franca, M.A. Gazziro, A.N. Ide, and J.H. Saito. 2005. A 3D scanning system based on laser triangulation and variable field of view. In *IEEE International Conference on Image Processing 2005*, Vol. 1. I–425. https://doi.org/10.1109/ICIP.2005.1529778 ISSN: 2381-8549.

[14] Camila Friedman-Gerlicz, Deanna Gelosi, Fiona Bell, and Leah Buechley. 2024. WeaveSlicer: Expanding the Range of Printable Geometries in Clay. In *Proceedings of the CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 352, 16 pages. https://doi.org/10.1145/3613904.3642622

[15] Devon Frost, Raina Lee, Eun-Ha Paek, and Jennifer Jacobs. 2024. SketchPath: Using Digital Drawing to Integrate the Gestural Qualities of Craft in CAM-Based Clay 3D Printing. In *Proceedings of the CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 344, 16 pages. https://doi.org/10.1145/3613904.3642684

[16] Madeline Gannon, Tovi Grossman, and George Fitzmaurice. 2015. Tactum: A Skin-Centric Approach to Digital Design and Fabrication. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. Association for Computing Machinery, New York, NY, USA, 1779–1788. https://doi.org/10.1145/2702123.2702581

[17] Madeline Gannon, Tovi Grossman, and George Fitzmaurice. 2016. ExoSkin: On-Body Fabrication. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. Association for Computing Machinery, New York, NY, USA, 5996–6007. https://doi.org/10.1145/2858036.2858576

[18] Ammar Hattab and Gabriel Taubin. 2019. Rough carving of 3D models with spatial augmented reality. In *Proceedings of the 3rd Annual ACM Symposium on Computational Fabrication (SCF '19)*. Association for Computing Machinery, New York, NY, USA, 1–10. https://doi.org/10.1145/3328939.3328998

[19] Mare Hirsch, Gabrielle Benabdallah, Jennifer Jacobs, and Nadya Peek. 2023. Nothing Like Compilation: How Professional Digital Fabrication Workflows Go Beyond Extruding, Milling, and Machines. *ACM Trans. Comput.-Hum. Interact.* 31, 1, Article 13 (nov 2023), 45 pages. https://doi.org/10.1145/3609328

[20] Nathaniel Hudson, Celena Alcock, and Parmit K. Chilana. 2016. Understanding Newcomers to 3D Printing: Motivations, Workflows, and Barriers of Casual Makers. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. Association for Computing Machinery, New York, NY, USA, 384–396. https://doi.org/10.1145/2858036.2858266

[21] Jonathan Keep. 2020. A Guide to Clay 3D Printing. http://keep-art.co.uk/Journal/JK_Guide_to_Clay_3D_Printing.pdf

[22] Jeeeun Kim, Clement Zheng, Haruki Takahashi, Mark D Gross, Daniel Ashbrook, and Tom Yeh. 2018. Compositional 3D printing: expanding & supporting workflows towards continuous fabrication. In *Proceedings of the 2nd Annual ACM Symposium on Computational Fabrication (SCF '18)*. Association for Computing Machinery, New York, NY, USA, 1–10. https://doi.org/10.1145/3213512.3213518

[23] Chen Liang, Anhong Guo, and Jeeeun Kim. 2022. CustomizAR: Facilitating Interactive Exploration and Measurement of Adaptive 3D Designs. In *Proceedings of the 2022 ACM Designing Interactive Systems Conference (DIS '22)*. Association for Computing Machinery, New York, NY, USA, 898–912. https://doi.org/10.1145/3532106.3533561

[24] Daniela Mitterberger, Selen Ercan Jenny, Lauren Vasey, Ena Lloret-Fritschi, Petrus Aejmelaeus-Lindström, Fabio Gramazio, and Matthias Kohler. 2022. Interactive Robotic Plastering: Augmented Interactive Design and Fabrication for On-site Robotic Plastering. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22)*. Association for Computing Machinery, New York, NY, USA, 1–18. https://doi.org/10.1145/3491102.3501842

[25] Ilan E Moyer, Samuelle Bourgault, Devon Frost, and Jennifer Jacobs. 2024. Throwing Out Conventions: Reimagining Craft-Centered CNC Tool Design through the Digital Pottery Wheel. In *Proceedings of the CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 347, 22 pages. https://doi.org/10.1145/3613904.3642361

[26] Alexei Pevzner, Saed Hasan, Rafael Sacks, and Amir Degani. 2020. Construction Operation Assessment and Correction Using Laser Scanning and Projection Feedback. In *Proceedings of the 37th International Symposium on Automation and Robotics in Construction (ISARC)*. International Association for Automation and Robotics in Construction (IAARC), Kitakyushu, Japan, 1247–1254. https://doi.org/10.22260/ISARC2020/0171

[27] 3D Potter. 2023. 3D PotterBot 10 PRO Real Clay 3D Ceramic Printer. https://3dpotter.com/printers/potterbot-10-pro

[28] D. Pye and E. Shales. 2008. *The Nature and Art of Workmanship*. Bloomsbury USA, New York, NY, USA.

[29] Raf Ramakers, Fraser Anderson, Tovi Grossman, and George Fitzmaurice. 2016. RetroFab: A Design Tool for Retrofitting Physical Interfaces using Actuators, Sensors and 3D Printing. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. Association for Computing Machinery, New York, NY, USA, 409–419. https://doi.org/10.1145/2858036.2858485

[30] C. Rocchini, P. Cignoni, C. Montani, P. Pingi, and R. Scopigno. 2001. A low cost 3D scanner based on structured light. *Computer Graphics Forum* 20, 3 (2001), 299–308. https://doi.org/10.1111/1467-8659.00522 _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/1467-8659.00522.

[31] Valkyrie Savage, Sean Follmer, Jingyi Li, and Björn Hartmann. 2015. Makers' Marks: Physical Markup for Designing and Fabricating Functional Objects. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. Association for Computing Machinery, New York, NY, USA, 103–108. https://doi.org/10.1145/2807442.2807508

[32] Pitchaya Sitthi-Amorn, Javier E. Ramos, Yuwang Wangy, Joyce Kwan, Justin Lan, Wenshou Wang, and Wojciech Matusik. 2015. MultiFab: a machine vision assisted platform for multi-material 3D printing. *ACM Transactions on Graphics* 34, 4 (July 2015), 129:1–129:11. https://doi.org/10.1145/2766962

[33] Blair Subbaraman and Nadya Peek. 2024. Playing the Print: MIDI-Based Fabrication Interfaces to Explore and Document Material Behavior. In *Extended Abstracts of the 2024 CHI Conference on Human Factors in Computing Systems (CHI EA '24)*. Association for Computing Machinery, New York, NY, USA, Article 267, 8 pages. https://doi.org/10.1145/3613905.3650966

[34] Jef Suina. 2024. Jeff Suina Art. https://www.jeffsuina.art

[35] Alexander Teibrich, Stefanie Mueller, François Guimbretière, Robert Kovacs, Stefan Neubert, and Patrick Baudisch. 2015. Patching Physical Objects. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. Association for Computing Machinery, New York, NY, USA, 83–91. https://doi.org/10.1145/2807442.2807467

[36] Timea Tihanyi. 2020. *Ceramic 3D Printing: Art and Mathematics*. Quality Press, Seattle, Washington. https://www.sliprabbit.org/book

[37] Mert Toka, Samuelle Bourgault, Camila Friedman-Gerlicz, and Jennifer Jacobs. 2023. An Adaptable Workflow for Manual-Computational Ceramic Surface Ornamentation. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST '23)*. Association for Computing Machinery, New York, NY, USA, 1–15. https://doi.org/10.1145/3586183.3606726

[38] Hannah Twigg-Smith, Jasper Tran O'Leary, and Nadya Peek. 2021. Tools, Tricks, and Hacks: Exploring Novel Digital Fabrication Workflows on #PlotterTwitter. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. Association for Computing Machinery, New York, NY, USA, 1–15. https://doi.org/10.1145/3411764.3445653

[39] Unfold Design Studio. 2009. Ceramic 3d Printing. http://unfold.be/pages/ceramic-3d-printing.html

[40] Joshua Vasquez, Hannah Twigg-Smith, Jasper Tran O'Leary, and Nadya Peek. 2020. Jubilee: An Extensible Machine for Multi-tool Fabrication. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3313831.3376425

[41] Leah A. Wasser. 2024. The Basics of LiDAR - Light Detection and Ranging - Remote Sensing | NSF NEON | Open Data to Understand our Ecosystems. https://www.neonscience.org/resources/learning-hub/tutorials/lidar-basics

[42] Christian Weichel, John Hardy, Jason Alexander, and Hans Gellersen. 2015. ReForm: Integrating Physical and Digital Design through Bidirectional Fabrication. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. Association for Computing Machinery, New York, NY, USA, 93–102. https://doi.org/10.1145/2807442.2807451

[43] 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 SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. Association for Computing Machinery, New York, NY, USA, 3855–3864. https://doi.org/10.1145/2556288.2557090

[44] Karl D.D. Willis, Cheng Xu, Kuan-Ju Wu, Golan Levin, and Mark D. Gross. 2010. Interactive fabrication: new interfaces for digital fabrication. In *Proceedings of the fifth international conference on Tangible, embedded, and embodied interaction (TEI '11)*. Association for Computing Machinery, New York, NY, USA, 69–72. https://doi.org/10.1145/1935701.1935716