

# BrickStART: Enabling In-situ Design and Tangible Exploration for Personal Fabrication using Mixed Reality

EVGENY STEMASOV, Ulm University, Germany

JESSICA HOHN, Ulm University, Germany

MAURICE CORDTS, Ulm University, Germany

ANJA SCHIKORR, Ulm University, Germany

ENRICO RUKZIO, Ulm University, Germany

JAN GUGENHEIMER, TU-Darmstadt, Germany and Institut Polytechnique de Paris, France

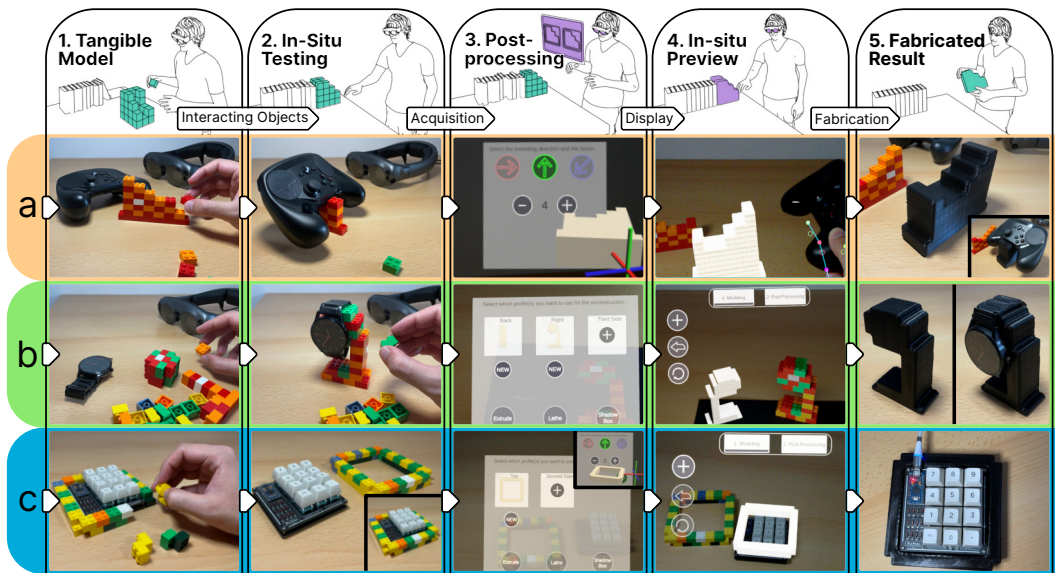


Fig. 1. BrickStART embraces in-situ tangible interaction to enable users to design personalized objects, such as stands (a – controller stand, b – watch stand) or enclosures (c – number pad case). Users first create a low-fidelity tangible model using interlocking construction bricks (1). It can be used to test for stability (2a), balance (2b), or fit (2c) early and in-situ. Users can easily alter this tangible model to fix errors they identified. Then, users scan the model with an AR headset to refine it (3), for instance, to smooth it (3a). The refined result can be previewed in-situ (4) and fabricated (5) to fulfill the user’s requirements.

Authors’ addresses: [Evgeny Stemasov](mailto:evgeny.stemasov@uni-ulm.de), [evgeny.stemasov@uni-ulm.de](mailto:evgeny.stemasov@uni-ulm.de), Ulm University, Ulm, Germany; [Jessica Hohn](mailto:jessica.hohn@alumni.uni-ulm.de), [jessica.hohn@alumni.uni-ulm.de](mailto:jessica.hohn@alumni.uni-ulm.de), Ulm University, Ulm, Germany; [Maurice Cordts](mailto:maurice.cordts@uni-ulm.de), [maurice.cordts@uni-ulm.de](mailto:maurice.cordts@uni-ulm.de), Ulm University, Ulm, Germany; [Anja Schikorr](mailto:anja.schikorr@alumni.uni-ulm.de), [anja.schikorr@alumni.uni-ulm.de](mailto:anja.schikorr@alumni.uni-ulm.de), Ulm University, Ulm, Germany; [Enrico Rukzio](mailto:enrico.rukzio@uni-ulm.de), [enrico.rukzio@uni-ulm.de](mailto:enrico.rukzio@uni-ulm.de), Ulm University, Ulm, Germany; [Jan Gugenheimer](mailto:jan.gugenheimer@tu-darmstadt.de), [jan.gugenheimer@tu-darmstadt.de](mailto:jan.gugenheimer@tu-darmstadt.de), TU-Darmstadt, Darmstadt, Germany and Institut Polytechnique de Paris, Paris, France.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

© 2023 Copyright held by the owner/author(s).

2573-0142/2023/12-ART429

<https://doi.org/10.1145/3626465>

3D-printers enable end-users to design and fabricate unique physical artifacts but maintain an increased entry barrier and friction. End users must design tangible artifacts through intangible media away from the main problem space (*ex-situ*) and transfer spatial requirements to an abstract software environment. To allow users to evaluate dimensions, balance, or fit early and *in-situ*, we developed BrickStART, a design tool using tangible construction blocks paired with a mixed-reality headset. Users assemble a physical block model at the envisioned location of the fabricated artifact. Designs can be tested tangibly, refined, and digitally post-processed, remaining *continuously in-situ*. We implemented BrickStART using a Magic Leap headset and present walkthroughs, highlighting novel interactions for 3D-design. In a user study ( $n = 16$ ), first-time 3D-modelers succeeded more often using BrickStART than Tinkercad. Our results suggest that BrickStART provides an accessible and explorative process while facilitating quick, tangible design iterations that allow users to detect physics-related issues (e.g., clearance) early on.

CCS Concepts: • **Human-centered computing** → **Human computer interaction (HCI)**; *Mixed / augmented reality*.

Additional Key Words and Phrases: Mixed Reality, Personal Fabrication, Tangible Interaction, Tangible Modeling, Tangible CAD, In-Situ Modeling, BrickStART

### ACM Reference Format:

Evgeny Stemasov, Jessica Hohn, Maurice Cordts, Anja Schikorr, Enrico Rukzio, and Jan Gugenheimer. 2023. BrickStART: Enabling In-situ Design and Tangible Exploration for Personal Fabrication using Mixed Reality. *Proc. ACM Hum.-Comput. Interact.* 7, ISS, Article 429 (December 2023), 29 pages. <https://doi.org/10.1145/3626465>

## 1 INTRODUCTION

Digital Fabrication can provide unrivaled, industry-grade precision to end-users and consumers, allowing them to create highly personalized physical objects. This is described in the vision of personal fabrication [9, 22, 23]. Proficient users may then design and fabricate decorative artifacts tailored to their taste (e.g., vases or ornaments), functional artifacts tailored to their mechanical requirements (e.g., mounts, attachments, spare parts), or any combination of those. In the future, given sufficiently powerful fabrication processes, *any* requirement for a physical artifact may be solvable quickly and satisfyingly through personal (digital) fabrication.

While personal digital fabrication makes these compelling promises (e.g., precision, personalization), it remains a tool used mainly by hobbyists (who were involved in – analog/digital – craft anyway) or technology enthusiasts (intrinsically motivated to engage with the technology) [33, 81]. This can be ascribed to various types of *friction* users have to overcome during design and fabrication: most design tools have to be learned, and involve complex, lengthy processes of design and iteration. In particular, design tools are used *ex-situ*, away from the problem space [6], are *intangible* (despite being used for creating *tangible* artifacts), and the exploration of physical consequences is left to either complex simulation software or fabricated iterations of the design (i.e., engaging in physical prototyping). We see a set of specific aspects to address here: 1) the design of (future / ultimately) tangible objects is mostly intangible; 2) the design of these future objects happens *ex-situ*, instead of *in-situ*; 3) users are restricted to exploring physical consequences either with a finished design (to simulate *digitally*) or after fabrication (to evaluate *physically*).

We propose BrickStART<sup>1</sup>, a mixed reality 3D-modeling tool supported by tangible elements (fig. 1), implemented for the Magic Leap One<sup>2</sup> head-mounted display (HMD). With BrickStART, users start their design by coarsely approximating their desired shape and features through colored, interlocking plastic bricks (e.g., LEGO®, fig. 1.1). Users may either build their entire design (fig. 1b) or define separate outlines (1 brick “deep”) to alter later on (fig. 1a). The brick model enables them to coarsely explore physical, tangible consequences of their design, such as the stability of

<sup>1</sup>Naming derived from Brick + Kickstart + AR

<sup>2</sup><https://ml1-developer.magicleap.com/en-us/home>, Accessed: 8.5.23

their object (fig. 1.1b) early in their design process. This also provides a worst-case estimate of fit (fig. 1.2c), balance (fig. 1.2b), or clearance. BrickStART offers a set of methods to alter and refine the design in mixed reality, like extrusion (fig. 1.3a), smoothing, and basic geometrical operations. Afterward, the design can be previewed in-situ (fig. 1.4) and altered further (e.g., smoothed). Lastly, users may fabricate the artifact they designed and use it within the physical context they explored and made their design in (fig. 1.5). With BrickStART, we focus on designing everyday household artifacts [76], over complex engineering challenges, as these household artifacts are an aspect of widespread adoption of personal fabrication and its embedding and increasing relevance in everyday life [81]. These household artifacts, such as jigs, mounts, or decorative vases, may not demand the highest precision, but benefit from users expressing their uniquely personal requirements, both functional and aesthetic, and especially in relation to their unique personal contexts. Established CAD (computer-aided-design) software, along with prevalent paradigms of 3D-modeling [81], do not necessarily fit this use-case, as they often focus on offering users the highest possible precision.

We explored the use of BrickStART in a user study ( $n = 16$ ). We found that users immediately got started designing the tangible model and were able to evaluate how a design would behave early on. This happened through tangible, hands-on explorations with the bricks and interacting objects, but also visually, using the digital model displayed in-situ through the HMD. Users with no prior 3D-modeling experience created fewer designs that did not comply with balance or clearance requirements using BrickStART, compared to Autodesk Tinkercad, indicating the suitability of the paradigm for novice 3D modelers. While BrickStART is not meant to provide users with unlimited precision and fidelity, it offers them a tangible, in-situ design process that is particularly applicable to everyday household items that may interact with existing counterparts in a physical environment [6, 51]. Such household items, like planters or stands for various objects, are well-represented in the popular designs in repositories like Thingiverse<sup>3</sup>. Unlike the use of Thingiverse or desktop-based CAD, BrickStART situates all *relevant* steps (i.e., design and exploration) of a personal fabrication process within the physical context of the future artifact and in conjunction with existing interacting objects, allowing users to test and iterate while designing (fig. 1.2). This enables more dynamic explorations and iterations of a design, without having to either 1) employ a fabrication device to *permanently* manufacture a prototype; or 2) transfer all information from the physical environment to set up a digital design and simulation, risking errors along the way. We argue that this is necessary to enable novices, inexperienced end-users [6, 51], or even non-users [81] to start exploring and engaging with personal fabrication in the future.

The contributions of this work are:

- **The concept of using tangible elements augmented with a mixed reality headset to define artifacts for personal fabrication.** This combination enables physical, hands-on exploration prior to fabrication while allowing users to continuously interact in-situ (i.e., in the object's future context). We further present the prototype implementation of BrickStART, a mixed reality modeling tool embracing this approach.
- **Insights gathered from a user study of BrickStART ( $n = 16$ ) and application walk-throughs,** not only confirming the feasibility of our approach but also showing that BrickStART may prevent novice users from making errors referring to measuring (clearance) and creating stable designs (balance) by allowing hands-on explorations without having to fabricate prototypes.

<sup>3</sup><https://www.thingiverse.com>, Accessed 2.5.23

## 2 RELATED WORK

BrickStARt is inspired by and related to 3 core directions: tangible computer-aided design (CAD) tools, 2D-to-3D CAD workflows, and in-situ design tools. Our focus lies on the *design* of artifacts and less on their fabrication, even if the brick assembly could be considered a fabrication method [60]. With *in-situ modeling or interaction*, we focus on the location of **future use** matching the location of design, not the location of fabrication. BrickStARt also aims to be an in-situ design tool that makes the context (i.e., interacting objects), but also the workpiece tangible. Ideally, a tool provides both tangibility and situatedness, to allow for quick in-situ design explorations [2], early testing, and easier transfer of requirements (e.g., sizes) between physical context and digital model [51].

### 2.1 Tangible CAD

Adding tangibility to computer-aided design tools introduces a perceptual dimension lost with most screen-based CAD systems. This enhances users' understanding of the physical properties a (future) artifact may exhibit. Prior works have explored tangible systems for both *design* and *fabrication*. Savage et al. allowed users to model in clay or other materials [74]. Stickers were used to annotate where the system should insert mounts for functional elements like buttons [74]. Jones et al. presented a comparable approach for interactive device design based on clay combined with placeholders for interactive elements [38]. CopyCAD by Follmer et al. enabled users to take real-world objects to scan and embed in digital designs [18], which was further explored with children as a user group in KidCAD [19]. Various approaches have previously focused on a "building-block"-like interaction for 3D design. This constrains the users' input to a fixed grid, in contrast to free-form approaches based on clay. StrutModeling by Leen et al. is a construction kit out of struts and hubs, enabling users to prototype objects in situ and in a tangible fashion [45]. However, their approach moves steps like postprocessing and previewing of post-processed models ex-situ (i.e., to a desktop computer). Anderson et al. presented a 3D-modeling approach based on tangible blocks or clay, which are scanned and interpreted to be rendered as a digital model [5]. Their work is predated by comparable approaches focused on architectural design [3]. Dynablock by Suzuki et al. presented "dynamic 3D printing", an approach to fabricate reconfigurable, low-fidelity 3D models [84], which allows for an overlap of design and fabrication. FaBrickation by Mueller et al. used building blocks to accelerate *fabrication* and therefore prototyping cycles [60], covering a different phase of the personal fabrication process than BrickStARt, which is focused on accessible design over quick prototype manufacturing. In the context of electronics design, ElectronicsAR supported novices designing circuits [17], and NatCut facilitated the design of enclosures [75] through tangible interaction and augmented information.

With BrickStARt, we similarly take a tangible approach to design for fabrication but want to situate all relevant steps of a design in the design's relevant physical context (i.e., where it will be *used*), unifying digital and physical spaces [15]. We furthermore embrace a lower fidelity and resolution, while still focusing on the objects' interactions with the physical environment. This tangible exploration with a low-fidelity model allows for early tests of physical properties. Meanwhile, actions involving leaving the block grid, altering the model further, and creating smooth surfaces are still enabled and explorable in-situ, in mixed reality.

### 2.2 2D-to-3D CAD Workflows

Both academic works and established CAD software have allowed users to define features in 2D to convert them to 3D features later (e.g., a 2D sketch that is extruded to create a 3D volume). To simplify the design process, BrickStARt also allows for this kind of workflow, in addition to the assembly of complete 3D objects. An influential example is Teddy by Igarashi et al., which inflates

2D outlines – drawn by novice users – into 3D models [34]. SketchChair by Saul et al. took a similar approach within the context of chair design [73]. Users began by sketching the desired outline and verifying the design through simulation afterward. From this design, a cutting plan for a 3D chair is generated. McCrae et al. presented FlatFitFab, which focuses on combining planar sections to 3D objects [52]. The tool further supports users by providing physics simulations, for instance, by accounting for loads put onto an assembly of planar sections. CutCAD is an open-source tool enabling users to create 3D objects based on assemblies of 2D parts [29].

Tools that focus on fabrication or overlap between fabrication and design have also been present in the literature. LaserStacker [88] and LaserOrigami [59] are both tools to create 3D geometry with 2D cutting machines, leveraging their outstanding speed in planar cutting. StackMold similarly focuses on fabrication, but uses stacked layers of 2D molds to create 3D objects [89]. HingeCore [1] and FoolProofJoint [62] similarly focus on the assembly of cut 2D elements to 3D objects.

In a similar fashion to these works, we want to benefit from 2D features for the design of 3D objects. We rely on metaphors established in CAD software, which allows users to *partially* define and explore objects and focus on relevant features. We furthermore want users to benefit from tangibility and an in-situ process while focusing on design instead of fabrication. Unlike FlatFitFab [52] or CutCAD [29], we aim to allow users to *explore* their design and its physical properties in a lower resolution, and tangible fashion which is quick to iterate and change without demanding material use.

### 2.3 In-situ Design and Fabrication Tools

We lastly consider BrickStART to be an in-situ design tool, which situates design and iteration within an artifact's future context. Comparable tools are particularly relevant in the context of design for fabrication. This allows users to tangibly experience both the workpiece and the context it will interact with after fabrication. Prior examples of in-situ *design* tools, for instance, include MixFab by Weichel et al. [94], which is a self-contained design environment using mixed reality. Users can introduce real-world objects into the system to alter geometry with their help [94]. Lau et al. presented Modeling in Context, which is a design tool leveraging tangible primitives for in-situ 3D-modeling [44]. Tangible Version Control by Letter and Wolf explored versioning of physical artifacts supported with an HMD [46]. DesignAR by Reipschläger and Dachsel [68] combined an HMD (head-mounted display) with an interactive surface, which for instance, allows users to trace reference objects and features. AutomataStage by Jeong et al. focused on creativity support in the context of designing automata [37].

In the context of (personal) fabrication, which focuses on the creation of tangible artifacts, in-situ *fabrication* tools are also present in literature, often combining the location of design with the location of fabrication. WireDraw presented an approach with instructions shown through an HMD guiding a user's manual fabrication process with a 3D-printing pen [101]. RoMA by Peng et al. couples design and fabrication, allowing manual input and fabrication by the user, with the option to introduce existing objects into the device's space [63]. Protopiper by Agrawal et al. can be considered a low-fidelity fabrication tool at a larger scale than most other approaches [2], and likened to tools for architecture-scale design support [53–55].

As an alternative to processes of design or fabrication, remixing [71, 80] and retrieval [82, 83] of existing (e.g., crowdsourced) designs have also been considered in prior works. CustomizAR by Liang et al. further expanded upon this notion by supporting users in discovering parametric designs and measuring real-world objects to customize the models [49]. In comparable fashion to greatly simplified modeling tools, they lower the effort required to learn and benefit from (3D-) modeling tools.



With BrickStARt, we embrace these notions but want to make both the user's model and interacting workpieces (i.e., the physical context) simultaneously tangible. This enables early testing and explorations, ideally reducing design errors that demand iterations. We furthermore want users to continuously interact in-situ with the object's future physical context, and not a decoupled standalone fabrication device or design workstation.

### 3 BRICKSTART

The following sections outline the design rationale for BrickStARt, our envisioned interaction flow for the tool, and describe the application itself. Based on the literature outlined in [section 2](#), we aimed for the following design goals: 1) **tangible interaction for modeling** [5, 31] that enables early checks of physics (in a quick, intuitive fashion [36]); 2) **fast explorations/iterations** before any digital design is attempted and before any permanent prototype is manufactured; 3) **continuous in-situ interaction** that does not require leaving a physical context while designing for it [82]. These aspects are meant to enable tangible, in-situ modeling, while simultaneously lowering the skill floor for 3D-design of physical objects (i.e., easy entry for novices [98]). We consider straightforward problems and similarly straightforward geometries to be particularly relevant for novices [4, 10, 50, 81], and want to emphasize that using desktop-based, "traditional" CAD tools for such tasks can prove to be a non-trivial task for genuine novices to the space [51]. Focusing on relatively simple geometries may not leverage the full potential of digital fabrication, but still provides meaningful benefits in terms of creative expression and the design and acquisition of physical artifacts that are uniquely tailored to an end user's requirements and context [67, 81]. With BrickStARt, we support this by employing low-fidelity tangible modeling, which enables early exploration of physical consequences, ideation, and iteration. This low-fidelity tangible model can then be refined and altered through a digital system, while still remaining in-situ (i.e., at the location of the artifact's future *use*), reducing the potential for errors and effort in transfers between digital design and physical context [51].

#### 3.1 Design Rationale and Interaction Flow

The following paragraphs elaborate on the design decisions we made for the development of BrickStARt, focusing on material choice, the hardware setup, and the intended context of use.

*Mixed Reality for Design.* For BrickStARt, we further rely on the means of spatial computing (i.e., a mixed reality head-mounted display – HMD). This allows BrickStARt to operate **continuously in-situ**: both in the tangible design stage, but also in the refinement stage later on. At no point in the design process, do users have to engage in transfers of requirements (cf., [6, 51]) to a (spatially) separate workspace location, but rather interact in-situ. This is meant to reduce friction, by either omitting or simplifying the transfers between physical (problem) space and (digital) design space. This includes measures (i.e., transferring an object's size to the design tool [51]), but also previewing (i.e., transferring a digital design to the problem space [82]). While both directions do not exhibit perfect precision, they can help detect or avoid errors early on, instead of doing so after fabrication, thereby saving material. We further consider spatial computing to be an emerging sensor platform, able to provide a variety of inputs to the design process (e.g., depth scanning, context understanding) in the future. Using a headset, specifically, further supports exploration/iteration: users can start working on a new brick design, while still seeing a previous, digital version of it – both situated in the same physical context. We consider the use of mixed/augmented reality the crucial part in this case, and less the specific hardware platform (e.g., HMD vs. phone), as long as they fulfill the outlined designed goals. Using a tablet or a phone to provide AR functionality would work in a similar fashion (e.g., as shown in [49]) and may yield different tradeoffs (e.g., occupation of hands, access to

required hardware). HMDs also provide stereoscopic depth cues, which are particularly influential at close range [14] and relevant when the device has to be hand-held while interacting [25]. We further combine gesture-based and controller-based inputs in our implementation of BrickStART (cf., [32]), to enable a separation of concerns between UI interactions and interactions made with 3D models in the application. The fundamental benefits of mixed/augmented reality are conceptually similar across these platforms and would stay the same, even with increasing miniaturization.

*Usage Context.* With BrickStART, we focus on a **subset of users** to support and a **subset of objects** to be designed: we focus on current non-users of digital/personal fabrication [81], or novice users [33]. Their requirements in the context of personal fabrication may differ from the ones put forward by expert users [27, 99], and especially from requirements of enthusiast users [33, 81]. For these users, searching for and customizing objects found on platforms like Thingiverse is a valid alternative, but not always possible or trivial to do [4, 81], as their intended object may not be available or customizable [4]. Similarly, customization of a parametric design to the user's requirements can similarly prove to be non-trivial, as users still have to *correctly* acquire and transfer requirements from the physical context to the design [6, 40, 49, 51]. In turn, this user group may engage with personal fabrication in a different way: focused on outcomes, and less on the process [33], and focused on small, everyday household artifacts [76, 90] instead of complex engineering challenges [81]. Geometry-wise, this implies that we focus on artifacts that, for the current version of BrickStART, are small-scale, do not require features below 8mm<sup>4</sup> in size, and are solutions to either mechanical/physical problems or fulfill aesthetic functions in users' households. We elaborate on the limitations of the approach in section 6, but emphasize that BrickStART is powerful enough for everyday household artifacts, as indicated by the study and the use-cases presented across sections 4.2 and 4.3.

*Material.* We settled on the use of interlocking plastic bricks (e.g., LEGO®), after considering alternatives like paper (to sketch on or to fold), malleable plastics (e.g., as used in [86]) or clay. Alternatives to these plastic bricks, such as modeling clay [74, 93], are present in the literature. However, interlocking plastic bricks provide a set of advantages, which we deemed appropriate for BrickStART: 1) The **learning effort** that is required to use them successfully, even to solve mechanical problems, is minimal – they can almost be considered a universal language for expression, that is even suitable for children [72], and their use has been proposed for learning CAD [100] as well. 2) The material properties are comparable to fabricated (i.e., 3D-printed) objects in terms of density and, by extension, physical behavior. While this similarity does not constitute fabrication-awareness [92, 95], it allows users to tangibly experience and understand how the fabricated result may physically behave and provide a worst-case estimate of this behavior. 3) The bricks are particularly **easy to reconfigure** and iterate with while maintaining appropriate structural stability. They are also inherently reusable without meaningful degradation over time.

*Process.* The envisioned process for BrickStART is outlined in figure 2. Our focus lies on the *design* of physical artifacts for personal fabrication, as this is the step where a user's personal input (i.e., their physical context and their requirements) is highly relevant. A user with access to personal fabrication technology (either through hardware at home, in public spaces like makerspaces [20, 21], or through a print service [11]) starts with a certain requirement or desire for a new, physical artifact. This requirement is formulated and evaluated within its own physical context (i.e., where it will reside after fabrication). The user then constructs a tangible model out of, in our case, interlocking building blocks (e.g., LEGO®). This tangible model is meant to be a lower-fidelity approximation of the desired artifact figure (fig. 2-1), or 2D profiles representing relevant geometric features.

<sup>4</sup>approximately half a brick wide

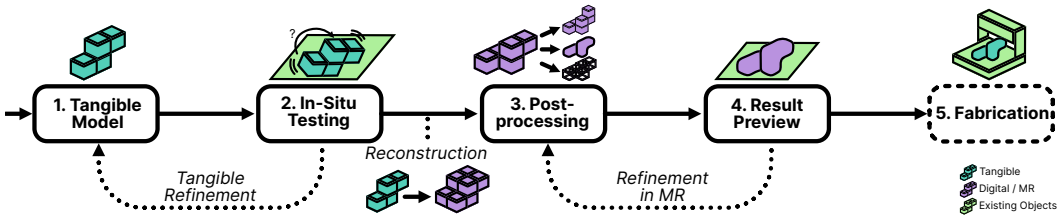


Fig. 2. Envisioned process for BrickStARt. Users start with a requirement and express it through an initial tangible model (1), which can be tested for physical aspects such as stability (2). This model is then reconstructed for further processing in AR (3). Altered results can similarly be previewed in-situ (4) and fabricated (5), if the result is deemed appropriate. With BrickStARt, we aim to avoid iterations with fabricated artifacts (e.g., 5  $\rightarrow$  2), but rather iterate and explore early and in-situ (2  $\rightarrow$  1 or 4  $\rightarrow$  3).

With this tangible model, the user can already evaluate certain aspects of their design: proportions with respect to the physical context, fit/clearance, or balance/stability (fig. 2-2). If the user notices flaws, the tangible model can be quickly altered, as afforded by the building blocks. After sufficient refinement, the user can now use BrickStARt to reconstruct the tangible model digitally. The user photographs the profile or object with the help of the HMD. From this photograph, 3D geometry is inferred and reconstructed. BrickStARt offers a set of methods to generate (3D) geometry based on a segmentation of 0-3 images. This geometry can then be postprocessed, for instance, to generate a smoothed surface (fig. 2-3). The resulting model can be further altered (e.g., by adding primitives like cubes) and previewed in-situ (fig. 2-4). If the user considers the result to be promising, the model can be handed off for fabrication. We do not assume any *specific* devices or services for fabrication to be particularly relevant for BrickStARt, but chose 3D printing as a suitable and currently available means for fabrication. Ultimately, a wide range of users may care more about results and less about the technological means enabling them [33, 81].

### 3.2 BrickStARt Application

BrickStARt was developed using Unity 2019.3 and Lumin SDK 0.24.1 for the Magic Leap One headset. Furthermore, the geometry3Sharp<sup>5</sup> and OpenCV for Unity<sup>6</sup> packages were used for geometry and image processing respectively. For the tangible building blocks, we initially settled on standard 2x2 interlocking plastic toy bricks. They are almost cube-shaped, with a size of approximately 15.8 x 15.8 x 11.4 mm, including the studs. Later, 4x2 and 8x2 were added, to reduce the influence of having a single block size on users' designs and approaches. The tangible building blocks cover the first phases of our envisioned process (fig. 2-1 & 2-2), while the AR application covers the later phases of the process (fig. 2-3 & 2-4). The transfer from the tangible model to a digitized one happens through an image processing pipeline, which we describe next.

**3.2.1 Image Processing.** We chose an image-based approach to digitize the tangible model. Current AR/VR headsets possess depth cameras for tracking. However, the ones in contemporary mixed reality devices generally deliver a mesh too coarse for the scale BrickStARt is working at. Users initially place their model on a black background and choose a "side" they are acquiring the object from (fig. 3-1). Then, the headset guides them to frame and photograph their model (fig. 3-2). The resulting picture is then shown to the user. If the user confirms that they have captured an appropriate view of their model (e.g., the model is within the frame and is not occluded or distorted),

<sup>5</sup><https://github.com/gradientspace/geometry3Sharp>, Accessed: 5.5.23

<sup>6</sup><https://enoxsoftware.com/opencvforunity/>, Accessed: 4.5.23



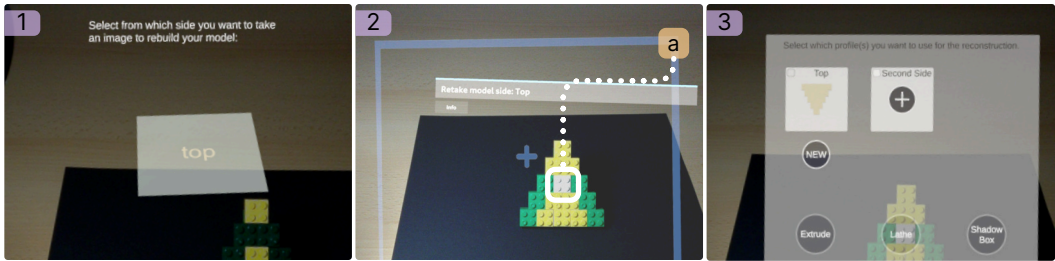


Fig. 3. To digitize the tangible model, the user frames it with the headset’s camera and takes a picture (1). BrickStART requires users to present the model on a black background and embed a single white brick in their model (2-a) to calibrate the grid size. The result is presented to the user (2). If the user confirms the photograph, the geometry is extracted and presented in a UI where further side-views can be captured (3), or re-attempted

the processing pipeline is started. The image is cropped to match the aspect ratio of the device’s field of view. It is then scaled down to a resolution of 300x255 pixels, to accelerate processing steps later in the pipeline. For the following steps, OpenCV for Unity 2.3.9 was used, which in turn uses OpenCV 4.3.0 internally. The scaled-down image is denoised and a Gaussian blur is applied. Afterward, mean shift filtering is used for image segmentation, which quantizes the number of colors<sup>7</sup>. To extract salient edges from the image, canny edge detection is used. The retrieved edges serve as a region of interest for the GrabCut algorithm [70], which separates the image into a foreground and background. To infer a scale of the tangible model a user has built, BrickStART requires them to use a single white brick in their construction (fig. 3-2a). The brick has to be visible once per photo (e.g., one brick visible from each perspective or multiple bricks, each visible from a single side only). This allows all other bricks to have any other color. Through this single brick, the  $x$  and  $y$  sizes of the brick grid are inferred. This reference brick could also be simply placed within the photographed frame or be added to the model and removed by the system later on. However, we found that using the reference brick as an actual part of the model avoided distortions introduced through the camera. After the size of the grid has been determined, BrickStART generates a binary bitmask from the image, based on possible (i.e., valid) locations of bricks within the captured frame. Based on this bitmask, a digital copy of the scanned tangible model’s outline is created (fig. 3-3). This enables users to apply specific reconstruction methods to create 3D models from the outlines.

**3.2.2 Reconstruction.** Having acquired one or more profiles (or sides of a model), users can now reconstruct it to a digital model to alter, postprocess, and preview it. BrickStART offers 3 distinct methods of reconstruction, present in industry CAD software: extrude, lathe, and triplanar. All methods are applicable to 2D outlines or profiles made with the bricks. Triplanar and extrude are additionally applicable to “solid” tangible models (i.e. if the user has built an entire object with the bricks). Creating 2D outlines allows users to focus on *relevant* features and generally performs better with respect to the reconstruction, as little to no occlusion is present. We embrace this 2D-to-3D workflow, as it allows users to explore and solve partial problems separately and combine them into a finished 3D design. We chose the 3 methods for a set of reasons: 1) they create useful geometry from 2D profiles, allowing users to solve partial and testable issues in 2-2.5D; 2) they are present in most, if not all, established toolchains; 3) they save users the effort of defining every single portion of the artifact, but rather “generate” geometry.

<sup>7</sup>As the total number is unknown (i.e., users are free to use any number of colors greater than 2 in their tangible model), k-means is not applicable here.

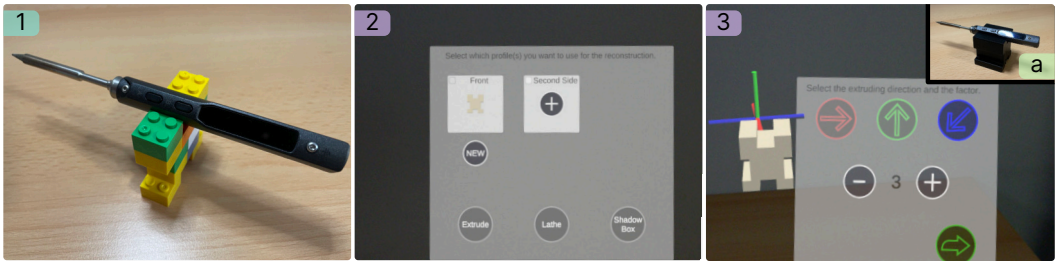


Fig. 4. Applying the *extrude* reconstruction method to a brick profile. The user has built a side profile of a stand for a soldering iron (1), which is photographed and reconstructed into a profile (2). Based on this profile, a 3D mesh can be generated. The user may choose an extrusion depth (3), and preview this result in-situ or fabricate it (3a).

The *extrude* reconstruction method uses a single 2D profile or side view and adds an arbitrary depth chosen by the user to it. This approach is common in established software, where features of a 2D sketch are selected and extruded to create a volume (e.g., `linear_extrude()` in OpenSCAD [96] or “Extrude” in Autodesk Fusion360 [8]). The process can be seen in figure 4.

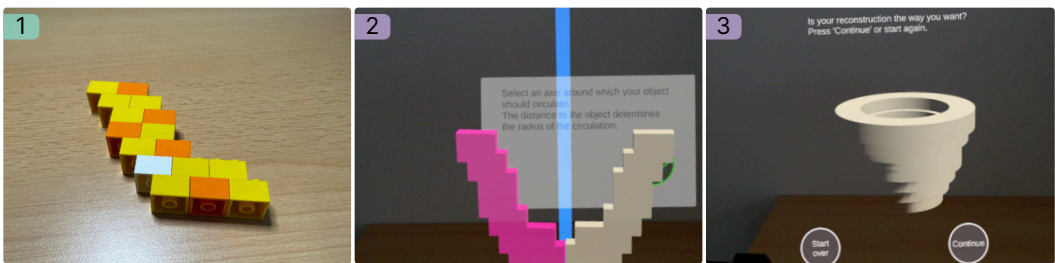


Fig. 5. Applying the *lathe* reconstruction method to a brick profile. The user has built a side profile of a vase (1), which is photographed and reconstructed to a 3D mesh, for which the user may choose an axis the profile is rotated around (2), and preview this result in-situ (3).

The *lathe* reconstruction method also uses a single 2D profile or a side view and rotates it around an axis to create a volume. It is likewise present in known 3D-modeling software (e.g., `rotate_extrude()` in OpenSCAD [96] or “Revolve” in Autodesk Inventor [7]). This process, as implemented in BrickStART, can be seen in figure 5.

Lastly, BrickStART offers users the *triplanar* reconstruction alternative. It is likewise inspired by functions offered by screen-based modeling tools (e.g., “Triplanar” in Nomad Sculpt [24] or “ShadowBox” in ZBrush [64]), but is a rarer occurrence<sup>8</sup> than *lathe* or *extrude*. It is present in tools focused on *sculpting* (i.e., more applicable to organic shapes), and less on “hard-surface modeling”, and is used for early definition of shapes. 2-3 profiles or side views are placed on orthogonal planes, extruded, and intersected to yield a 3D object. While it is not a complete 3D scan of an object, it creates a fitting representation based on profiles or side views. 2 profiles (e.g., top and side view) generally suffice to define a complex 3D object. The process can be seen in figure 6.

<sup>8</sup>openSCAD offers an example of this approach, but no built-in function: <https://files.openscad.org/examples/Advanced/GEB.html>, Accessed: 1.5.23

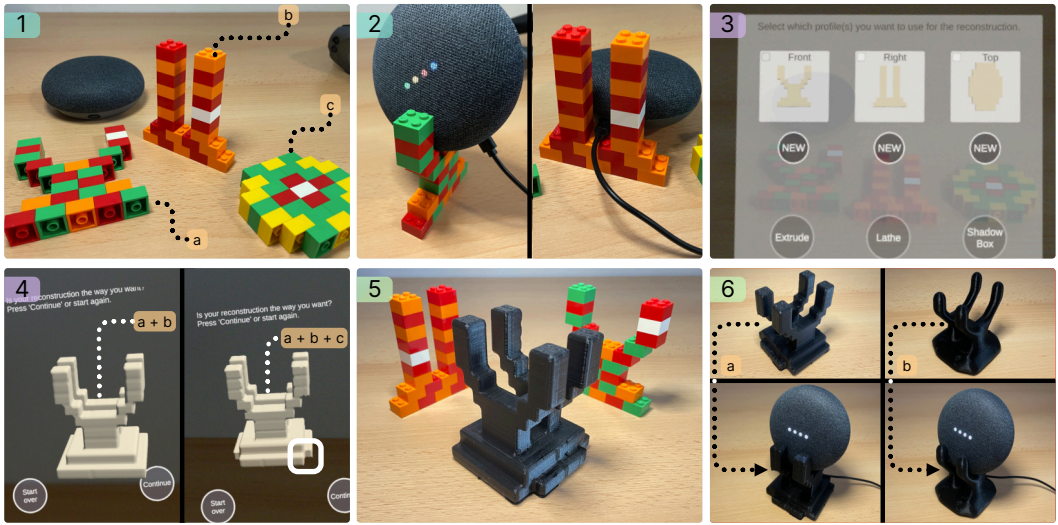


Fig. 6. Applying the *triplanar* reconstruction method to a set of brick profiles. The user has built 3 profiles, depicting a stand for a smart speaker from three axes (1). Profile “a” represents the main holding geometry, profile “b” is an opening for the wire, and profile “c” represents the footprint of the setup (2). All are acquired with BrickStART (3). Combining 2 profiles already yields a usable result; adding the third one refines the outline further (4). The printed result mimics the initial profiles (5) and holds the speaker without (6a) and with postprocessing (6b), as intended.

**3.2.3 Modeling and Previewing.** After having reconstructed the desired model digitally, users can start manipulating it in AR. This offers further opportunities to ‘leave’ the initial fixed grid of tangible building blocks. Users are able to freely scale the digital model. They can likewise rotate and move it, which allows them to preview the (altered) model in-situ, to verify aesthetic fit within the physical context, and visually estimate functional interaction [82].

BrickStART also provides functionality for geometric primitives. Users can choose and add from a set of geometric primitives (e.g., cubes or spheres). These primitives can be merged with the reconstructed model to compensate for features users may have missed with their tangible model or features they were unable to model with the tangible bricks. This also helps users who want to circumvent the fixed grid and scale specified by the bricks initially.

**3.2.4 Postprocessing.** Lastly, BrickStART offers users 2 ways to postprocess their models: smoothing and a conversion to a lattice structure (fig. 7). To match other objects in the users’ environments, BrickStART allows users to smooth the model they have made and scanned (fig. 7a). This compensates for the blocky look (cf., [43]) introduced by the tangible building blocks, smoothing out meshing artifacts, too. We opted for a low degree of user control to simplify this step, applying uniform smoothing across the entire model while retaining the most distinctive features. Despite this approach, the benefits of the tangible building blocks remain: they may still be a tangible worst-case estimate of fit or clearance. Figure 7 depicts a candle holder made with the tangible building blocks (fig. 7.1). The candle fits inside the cavity left open for it. After smoothing, this ‘fit’ remains (fig. 7b.3), albeit with slightly more clearance. This is also the case for the holder where lattice postprocessing was applied to yield interesting shadow patterns (fig. 7a.3).

BrickStART also offers users the possibility to convert the finished model to a lattice-like structure. The result of this operation can be seen in figure 7a. We consider this to be an additional, mostly



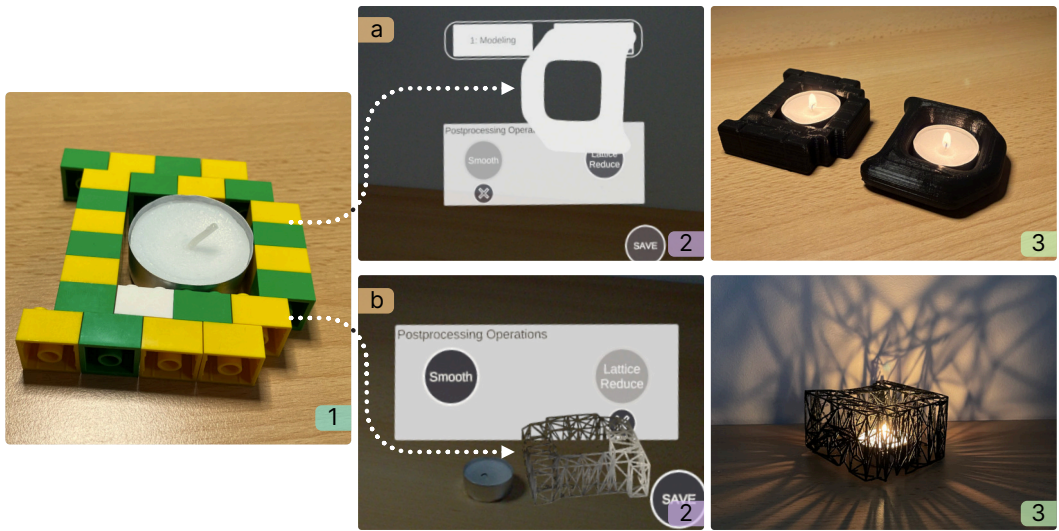


Fig. 7. Postprocessing options for a candle holder: uniform smoothing (a) and conversion to a lattice (b). 1) original brick design with candle; 2) digital, in-situ preview; 3) fabricated results.

visual postprocessing step. However, it may also offer more functional benefits: it is possible to use the resulting structure to hold objects (e.g., as a pencil holder or a trellis for plants). Furthermore, this step may save material for additive manufacturing (if printed support-free), and potentially accelerate the fabrication process, as demonstrated by Mueller et al. [58]. Lastly, users may choose to save their finalized model as an .obj file.

## 4 EVALUATION

To confirm the feasibility of BrickStART and to explore how users may interact with it, we conducted two user studies in total: an initial pilot study with 6 participants, and a more in-depth evaluation with 16 participants. We furthermore evaluated BrickStART through walkthroughs, to further probe and demonstrate its capabilities and limits. The following sections describe these evaluations.

### 4.1 Pilot User Study

We first conducted a smaller-scale pilot study with BrickStART. This pilot study was meant to elicit feedback on the concept and detect initial issues with the tool.

*Sample and Procedure.* We recruited a set of participants relying on convenience sampling [16, 30]. Out of the 6 participants in the pilot study, 2 identified as female, and the remaining 4 participants identified as male. Their age ranged from 24 to 31 ( $\bar{x} = 27.67$ ). All participants were students and had at least some experience with 3D modeling tools, all of them had used interlocking building block toys before (e.g., LEGO®), and 5 of 6 had used an AR headset before. During the study, participants had to solve 2 tasks. Task 1 involved creating a vase with BrickStART, to familiarize themselves with the tool. The second task was meant to be more challenging or at least offer more room for complexity: Half of the participants were tasked with modeling a bookend; the other half had to create a coat hook with two separate hooks.

*Results.* The 12 resulting objects are depicted in figure 8. They consist of 6 vases (fig. 8a), 3 bookends (fig. 8b), and 3 coat hooks (fig. 8c). We found that users were able to quickly grasp

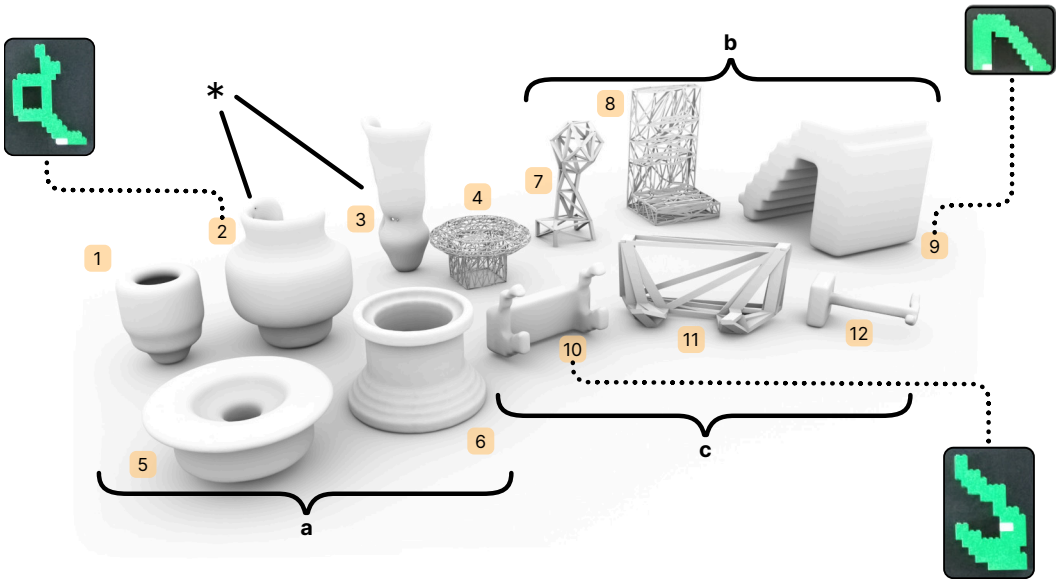


Fig. 8. Results of the 6 users built during the first study: 6 vases (a), 3 bookends (b), and 3 coat hooks (c). The 2 models denoted with an asterisk (\*) exhibit a faulty mesh from the reconstruction process. Select 2D brick inputs are depicted in green.

BrickStARt and use it to express both functional and, to a degree, aesthetic requirements. BrickStARt was rated with an average score of 2.67 for the PSSUQ questionnaire [47] and reached an average SUS score of 68.33. This was both observed by the experimenter and reflected in the comments made by the participants. Furthermore, the use of single-size bricks brought an unintentional design constraint, as users were forced to build their models in staggered layers. This is, for example, visible in Figure 8, models 9 and 10 and their respective brick profiles made by the participants. Lastly, we found the triplanar reconstruction appreciated by participants, but rarely used for the tasks given to them, despite it being applicable to both the bookend and the coat hook tasks. In the terminology of Resnick et al. [69], this indicates a higher skill floor than intended despite offering a potentially broad range of possible geometries (wide walls [69]). In turn, the lathe and extrude functions were well-received and required little explanation, hinting at a low skill floor [69].

*Takeaways for Main Study.* The core limitations of the pre-study were 1) the high amount of experience of the participants with respect to CAD software; 2) the absence of a baseline to compare BrickStARt with, and 3) the fact that it focused on objects that interact with their environment physically, but these counterparts (e.g., **plant**  $\Leftrightarrow$  vase) were not part of the setup and task. We furthermore changed our approach to the provided bricks: in the second study, users were encouraged to use multicolored brick assemblies to support recognition, and we added longer bricks to the users' selection to reduce the influence of having only one brick size available on the design. The lighting in the study environment was also improved to support the algorithm further. In addition, a set of minor issues with the implementation (e.g., how users grab and reposition objects in space, unresponsive buttons) were rectified.



## 4.2 User Study

The follow-up study was conducted to engage more deeply with the concept of tangible, in-situ 3D modeling by actively taking into account interacting objects from the physical environment (i.e., *context*) and comparing BrickStARt with a more established, novice-friendly 3D modeling tool: Autodesk Tinkercad<sup>9</sup>.

### 4.2.1 Procedure and Sample.

*Study Design.* The study followed a 2x2 within-subjects design, combining the factors tool (levels: Tinkercad and BrickStARt) and task (levels: phone stand and vase), with participants designing one model with each tool. The order of tasks and tool combinations was counterbalanced, and we ensured an even distribution of combinations and orders using a Latin Square to avoid carryover effects. Following the insights from the pilot study, we chose the 2 tasks based on the following criteria: 1) meaningful physical interaction with the environment (i.e., the phone and the potted plant), 2) simple enough geometry to be designed within 10-15 minutes, and 3) the potential for aesthetic expression of the participants. When observing the most popular designs on Thingiverse<sup>10</sup>, phone/tablet stands and vases are a reasonably common type of design (in addition to mechanical parts, organization tools, and various figurines). For the alternative tool, we settled on Tinkercad, established in the personal fabrication community as a particularly accessible 3D-design tool [12, 56]. Novices had to quickly learn 2 new tools over the course of the study, and both Tinkercad and BrickStARt can, arguably, be considered suitable for that.

*Metrics.* We acquired a set of qualitative and quantitative metrics (i.e., dependent variables) for the evaluation. Primarily, we were interested in users' success rates and the specific errors they may make in case of failure. We formalize "success" primarily based on function: users were tasked with "making a phone stand" and "making a vase". To further guide the users, each task consisted of 2 core objectives and an optional one. Users were handed printed cards that summarized the objectives and could reference them anytime. The phone stand task asked for "an angle of approximately 50°-60°", the possibility for it to "hold the phone in vertical and horizontal orientation", and, as an optional objective, "allow a charging cable to pass through". The vase task asked for "less than 1cm of play" between the inner and outer pot, a design that does not "topple easily", and optionally, a visually interesting feature on the vase. These aforementioned criteria can be formalized under the functionality of the object. In addition to that, the stability (i.e., balance) and the fit (i.e., with respect to the interacting object) were relevant rating dimensions. The time participants took to "solve" (following their own judgment) a task was recorded. We further recorded the raw NASA TLX (Task Load Index) [28] and the SUS (System Usability Scale) [13] after each task, while also presenting sets of non-standardized statements to be rated on Likert-Scales. Lastly, we solicited comments from all participants, specifically asking for positive and negative aspects of the respective tools. To better understand users' processes, their actions were recorded on video after explicit consent – Tinkercad tasks were recorded using OBS<sup>11</sup> on a laptop computer and BrickStARt tasks were recorded from a smartphone pointed at the participant's working area (a table).

*Participants.* We recruited 16 participants, across a broader range of users than in our pilot study. Out of the 16 participants, 7 identified as female, and 9 participants identified as male. Their age ranged from 18 to 33 ( $\bar{x} = 23.63$ ). 5 participants have used 3D modeling software before, and all 16 had experience with building blocks. 9 had used an AR headset before, but did so for a total

<sup>9</sup><https://www.tinkercad.com/>, Accessed: 2.5.23

<sup>10</sup><https://www.thingiverse.com/>, Accessed 2.5.23

<sup>11</sup>Open Broadcaster Software, <https://obsproject.com/>, Accessed: 1.5.23

duration below 10 hours (e.g., in the context of other user studies). The participants were mostly students with various backgrounds (e.g., computer science, biology, finance, mathematics).

*Procedure.* A general introduction regarding the study procedure was done first. Participants had to do 2 tasks in total: a phone stand and a vase, either using BrickStARt or Tinkercad. Prior to each task, participants received a brief introduction to the tool and the criteria relevant to their task. Similarly, they received a brief introduction to 3D-modeling for fabrication as such. For BrickStARt, they were introduced to the process (outlined in Figure 2), as implemented in the system, first. They were then introduced to fundamental interaction techniques for the HMD (i.e., controller inputs and gestures) and the general interaction flow of BrickStARt. For Tinkercad, users received a similar introduction involving all fundamental functions needed to achieve their goals (e.g., object manipulation, translations, primitives). Similarities between the tools were mentioned (e.g., the presence of Boolean operations) where fitting. For both introductions, users were allowed and encouraged to use and explore the system for up to 5 minutes, familiarizing themselves with the respective tool. Users were allowed to ask questions about functions at this stage. They were informed that their design was meant to be 3D-printed, and would be made out of plastic. Interacting objects were available to the users: a phone (LG Nexus 5X,  $147 \times 73 \times 7.9\text{mm}$ ) and a small houseplant ( $D_1 = 50\text{mm}$ ,  $D_2 = 38\text{mm}$ ,  $h = 50\text{mm}$ ). For the Tinkercad tasks, users were also given a caliper and a ruler to measure these interacting objects. Participants were suggested to take 15 minutes to complete their design, but there was no hard cutoff after these 15 minutes, letting all participants finalize their design. On average, the study took between 50 and 70 minutes. The tasks themselves took between 13 and 32 minutes using BrickStARt<sup>12</sup> and between 7 and 15 minutes using Tinkercad. All participants were reimbursed 11 € for their time.

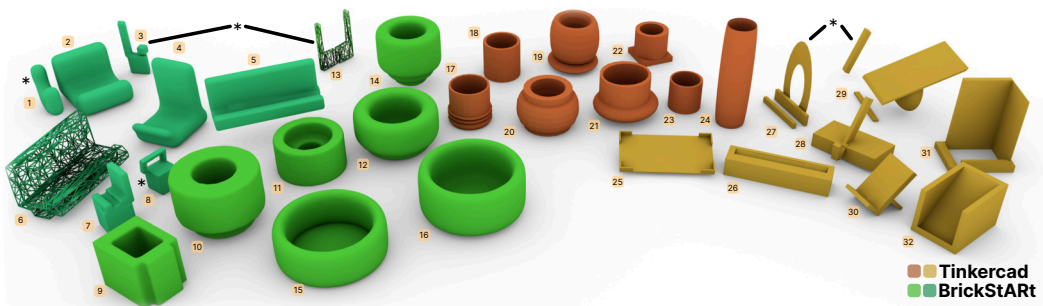


Fig. 9. Models made by the participants in the second user study: vases made with BrickStARt (light green) and Tinkercad (orange), and phone stands made with BrickStARt (teal) and Tinkercad (yellow). Models denoted with an asterisk (\*) exhibit a faulty mesh (i.e., one that might be manufacturable, but would not function due to its geometry) from the scanning process (in BrickStARt) or from design errors (in Tinkercad).

**4.2.2 Results.** The models made by the participants are depicted in figure 9 – 16 vases and 16 phone stands. Out of the 32 models made, 11 models completely fulfilled their intended function, 15 did so at least partially, and 6 did not. When split by the tool used, models made with Tinkercad were non-functional 2 times, partially functional 9 times, and functional 5 times. In contrast, models made with BrickStARt were non-functional 4 times, partially functional 6 times, and functional 6 times. Completely non-functional results emerged either through issues during scanning with

<sup>12</sup>the longer durations are partly explained by technical issues, but also due to some users strongly focusing on their brick model

BrickStARt (fig. 9: 1, 3, 8, 13), or through errors made by the users while modeling with Tinkercad (fig. 9: 27, 29).

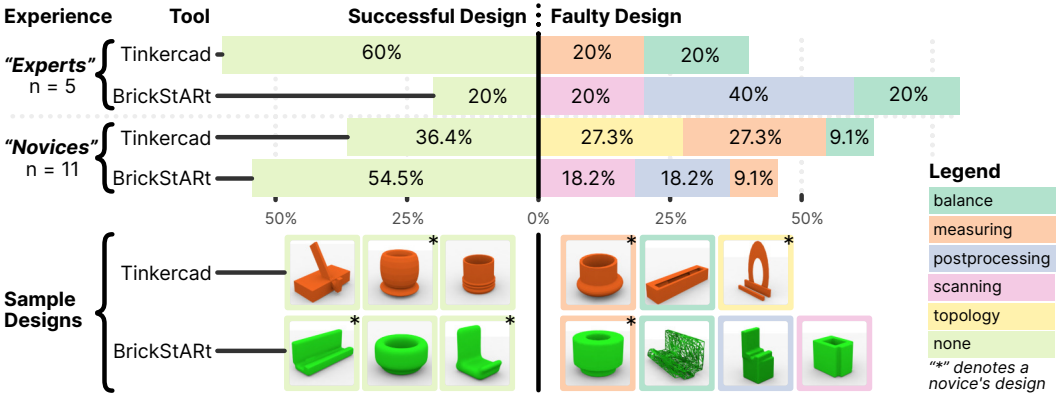


Fig. 10. Core design issues split by prior CAD experience (column 1) and tool used (column 2). Faulty designs are further classified by their core issue (color coding, see legend). Below: samples of designs and design issues made with the tools, first row – Tinkercad, second row – BrickStARt (outline color depicts core issue of design, asterisks denote a novice user).

**Error Analysis:** With the finished designs, we proceeded to conduct a more in-depth analysis of errors and issues in the models. 3 core categories were used to evaluate the designs: **fit** (whether the plant or phone fit), **balance** (whether the object would stand on its own), and **function** (whether it fulfills the core function as intended). Each design was scored with respect to the 3 categories on a discrete scale of “yes”, “partial”, “no”. Designs with deficiencies were then labeled by 2 of the authors to determine the core issue of a design. Comments regarding the function and errors were gathered initially and then converted to labels representing recurring themes. For non-obvious issues (i.e., ones where one would either have to set up a simulation to verify them, or fabricate them), or disagreements between the 2 raters, the models were 3D printed with Cura’s default settings in PLA filament and tested manually with the interacting objects users relied on in the study (e.g., fig. 9: 7, 11, 24, 26, 30). While evaluating failure reasons, we formalized 5 recurrent reasons for the users’ designs: balance, measuring, topology, postprocessing, and scanning (the latter 2 being specific to BrickStARt). The distribution of issues is depicted in figure 10. **Balance** issues were rated for objects that are unable to stand on their own (e.g., fig. 9: 6, 30), or would be easy to topple using a light touch (e.g., fig. 9: 14). **Measuring** issues arose when users either measured incorrect values, or transferred them incorrectly to the design environment, leading to incorrect clearances (e.g., fig. 9: 25, 32). **Postprocessing** issues were unique to BrickStARt and describe the digital alteration of a design that may have been functional, but was rendered unusable after scaling (e.g., fig. 9: 7), or applying other postprocessing operations (e.g., fig. 9: 14). **Scanning** issues were similarly unique to BrickStARt, and describe a class of problems where BrickStARt was unable to reliably reconstruct the intended design from the scanned brick model (e.g., fig. 9: 1, 3, 8, 13). **Topology** issues describe the design of non-functional or non-practical geometries, for instance, are missing connecting elements to work (e.g., fig. 9: 27, 29).

For the **overall function**, the success rate was comparable (*BST* : 7, *TK* : 7), while partial functionality was more often achieved with Tinkercad (*BST* : 2, *TK* : 4). For **stability**, a similar picture emerged, with Tinkercad performing slightly better (stable: *BST* : 9, *TK* : 10, unstable

*BST* : 3, *TK* : 2). This was also the case for **fit** (fits: *BST* : 4, *TK* : 4, does not: *BST* : 7, *TK* : 8). While the overall failure rate of both systems was fairly similar, we conducted a deeper labeling of the different *types* of errors. We found that the majority of errors in BrickStART occurred due to current implementation issues (e.g., scanning 18.8% or postprocessing 25%), and that balance or clearance issues were a rarer occurrence in BrickStART, compared to Tinkercad (*BST* : 12.4%, *TK* : 37.5). This has, however, to be considered under the assumption that the system failures in BrickStART would have yielded impeccable designs, which is not necessarily the case. Additionally, 3 designs made with Tinkercad failed due to incorrect topology (i.e., disconnected parts – fig. 9: 27, 29, 31).

**Prior experience** with 3D modeling seemed to have an effect on which errors users made and whether they made any at all (fig. 10). With no prior experience (i.e., *novices*), 6 users succeeded with BrickStART, compared to 4 successes with Tinkercad. Only 1 of the novices made measurement/clearance errors with BrickStART, compared to 3 using Tinkercad, and no novice using BrickStART misjudged balance. Novices using Tinkercad also made errors in topology, forgetting to connect elements of their models, which was not the case for BrickStART. In turn, users with prior experience generally performed better with Tinkercad, with 1 occurrence of measurement error and 1 balance issue. With BrickStART, 3 failures were due to technical issues, and 1 due to misjudged balance, emerging after smoothing.

*Questionnaire Data:* All participants agreed that using the bricks was helpful for their design process and generally agreed that with both tools, they were able to estimate sizes (*BST* : 62%, *TK* : 88%) and proportions (*BST* : 75%, *TK* : 69%). Similarly, a majority of participants (81%) agreed that both Tinkercad and BrickStART allowed them to explore how their object would look like after fabrication. Tinkercad achieved a higher average SUS score than BrickStART (75.6 vs. 53.6), which primarily originated from technical issues, such as imprecise reconstruction and errors in postprocessing leading to frustration. This explanation was further supported by the comments left by the participants. 44% of users would have liked to 3D-print their design after making it with BrickStART, compared to 35% of the Tinkercad users. In contrast, only 12% of the users rated their design as “pleasing” for BrickStART, compared to 38% for Tinkercad.

When asked for which tasks they would prefer either of the 2 tools (on a 5-point-scale), users preferred BrickStART for objects where they want to understand physics (*BST* : 56%, *TK* : 12%), objects that do not require precise measurements (*BST* : 56%, *TK* : 19%) and objects for which they may not immediately know a design approach (*BST* : 44%, *TK* : 31%). Tinkercad, in contrast, was preferred for objects demanding precision (*BST* : 0%, *TK* : 88%), objects where the solution is known (*BST* : 19%, *TK* : 75%), and complex objects (*BST* : 19%, *TK* : 75%). These patterns changed slightly when observing novices compared to experienced users: 60% of users with prior experience leaned towards BrickStART for which they do not immediately know a solution to, and 80% of them leaned towards BrickStART for objects that do not require precise measurements. Prior experience with AR had an effect on tool preference regarding the *looks* of objects. For “objects for which looks matter”, users with prior AR experience preferred BrickStART (*BST* : 78%, *TK* : 22%), while users without prior AR experience leaned towards Tinkercad (*BST* : 14%, *TK* : 71%). This pattern repeated for the contrary of the question (“objects for which looks do not matter”): users with prior AR experience preferred Tinkercad (*BST* : 22%, *TK* : 67%), while users without prior AR experience leaned towards BrickStART (*BST* : 57%, *TK* : 14%). The raw NASA TLX exhibited little difference between the tools, except for the physical demand and the temporal demand scales. BrickStART exhibited a higher physical demand than Tinkercad. Users also generally reported a lower temporal demand while using BrickStART than Tinkercad.

*Comments and Observations:* BrickStART was acclaimed for the intuitivity it had during the tangible modeling phase (“*handy*” – P2, ). Tangible modeling was further praised for being “... *playful*”

[and] ... *making the start of modeling rather easy*” (P13). For all participants, we observed them testing their design with their brick model combined with interacting objects (i.e., plant, phone) at least once prior to scanning. This approach to defining separate features to extrude/revolve later was also appreciated (“... *efficient and intuitive*” – P5, “... *practical*” – P7). The AR aspects of BrickStART were similarly well-received, as it was “*very helpful for roughly estimating sizes*” (P10), BrickStART was “*more visual*” (P8) and that “... *the visualization helped embed the size of the phone in the process*” (P16). These benefits echo observations during the procedure, where users actively embedded the environment and interacting objects in their design process, leveraging the spatial nature of the interface and process during the AR phase, *continuously interacting in-situ*. Negative comments referring to BrickStART were primarily due to issues in scanning (P1, P8, P11) or during postprocessing (P11, P13), which have affected user experience negatively. Users also were missing features for fine-grained manipulation (P5, P10). Tinkercad was similarly well-received, as it was “*Very easy to estimate sizes and proportions, easy to fix mistakes quickly*” (P3), and it provided “*Precise rotation and scaling tools*” (P11). However, it was also criticized for view controls (P7, P10, P12, P15) and issues while combining objects or features – “*It was somewhat unclear if there are gaps or not between two objects*” (P9). This was relevant for several of the models (e.g., fig. 9: 19, 25, 29).

### 4.3 Walkthroughs

The following sections depict three walkthroughs through tasks solvable with BrickStART. With these use-cases, we want to expand upon the user studies, by providing a more detailed view of the potential capabilities of BrickStART, highlighting approaches that were not feasible to test in the short timeframes in either of the 2 studies.

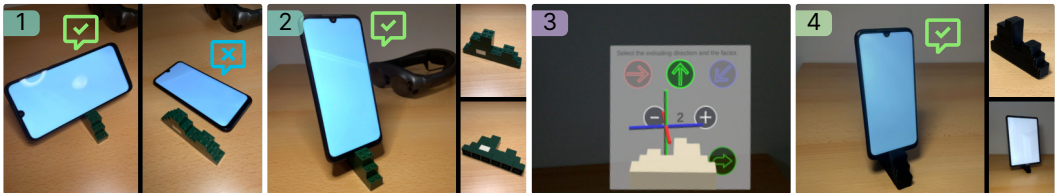


Fig. 11. A user’s goal is to create a phone stand that works for vertical and horizontal orientations of the device. 1) initial model, partially fulfilling the requirement; 2) further bricks are added for stabilization; 3) acquisition and extrusion of the profile; 4) fabricated result, holding the phone as expected and a tablet beyond that.

**4.3.1 Phone Stand – In-situ Physics Exploration.** BrickStART enables users to explore physical consequences in their model early in the design process. Iteration then happens while using reconfigurable, reusable tangible building blocks, instead of fabricating a prototype (e.g., 3D-printing it), testing it, and resolving issues that were detected. In figure 11, a user wants to create a simple phone stand. The stand is supposed to hold the user’s phone, while allowing it to be oriented vertically (for video calls) or horizontally (for watching videos), akin to the task in study 2. The user creates an initial model, which works for the horizontal orientation (fig. 11-1). After re-orienting the phone, the construction topples. The user then adds additional bricks (fig. 11-2). Notably, the user can test the model for stability in 1 axis, despite being essentially “flat” (i.e., one brick deep). The corrected and validated design is now scanned and extruded to match the phone’s width (fig. 11-3). The resulting model can be previewed in-situ and altered further, if needed. The fabricated result (fig. 11-4) fulfills the initial requirements and may even hold larger devices, which indicates



how the bricks serve as a “worst-case-estimation”, albeit one that may obscure the full capabilities of the design.

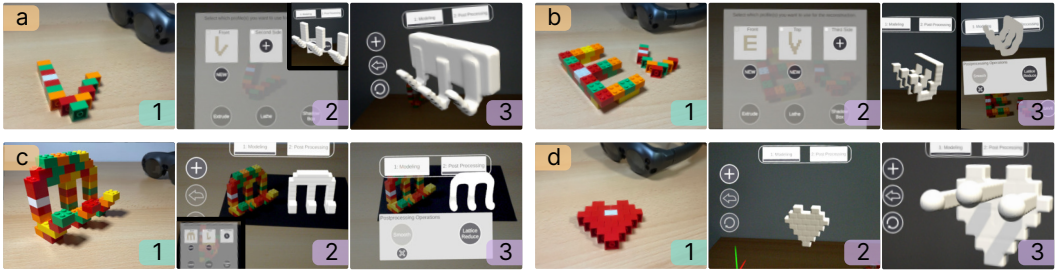


Fig. 12. 4 approaches to the creation of a three-pronged coat hook with BrickStART. a) multiplying an initial design; b) triplanar reconstruction from separate profiles; c) triplanar reconstruction from a solid model, smoothed in postprocessing; d) acquired backplate combined with primitives.

**4.3.2 Coat Hook – Varying Paths to Success.** Figure 12 depicts 4 different approaches to the creation of a three-pronged coat hook. The first approach (fig. 12-a) allows users to focus on the most important part of the object: the hook itself. Users initially build a tangible model of a single hook, acquire, and extrude the scanned design. Afterward, users can duplicate their design, arrange the resulting hooks, add a cube to serve as a backplate, scale it to match the distance between the hooks, and combine the 3 objects. The second approach (fig. 12-b) depicts a user creating 2 profiles instead of one. One depicts a single hook itself, the other describes the connecting geometry between the 3 hooks. As before, the outlines are acquired by BrickStART, combined through the triplanar reconstruction, and presented to the users for in-situ previewing. Lastly, the users may also smooth the result, which yields a more organic shape. The third approach (fig. 12-c) involves a user creating the *entire* hook with their tangible model. The model is photographed from 2 sides and, as before, the triplanar reconstruction is applied. This yields a mostly faithful recreation of the original model. The fourth approach (fig. 12-d) involves a user focusing more on the decorative geometry of the hook’s backplate. This backplate is built first, using the tangible building blocks. It is acquired with BrickStART and extruded to a default depth (i.e., one that is one brick deep). The users then add elongated cubes to the plate and merge them into a single model, resulting in a functional hook. This approach treats BrickStART more like a primitive-based 3D-modeling tool, as the most (functionally) relevant features of the hook are created digitally only.

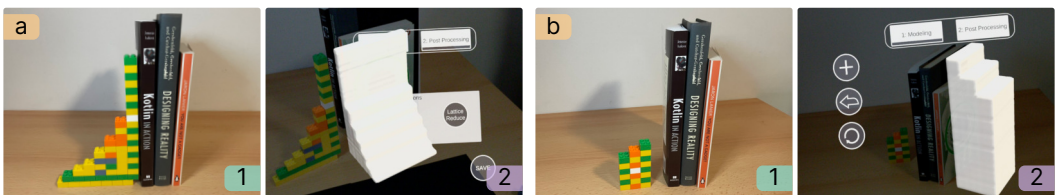


Fig. 13. Creation of a bookend with BrickStART while dealing with its low spatial resolution. Using a to-scale tangible design (a). Creating a tangible design that is not-to-scale, and postprocessing + scaling it to fit in AR (b).

**4.3.3 Bookend – Dealing with Scale & Resolution.** The use of the tangible building blocks enforces a certain resolution and therefore also a certain angularity. This is appropriate and beneficial

for early ideation and iteration, but may not be desirable for a finished, fabricated object. Using BrickStART, users are able to compensate for this in three ways: by creating a *larger* tangible model than necessary, creating a *smaller*, or by smoothing a to-scale model in AR later on. Figure 13 depicts this for the task of creating a bookend. One option is creating a to-scale tangible model (fig. 13a). This enables the exploration of proportions early on. Alternatively, users may create a tangible model that is not-to-scale (fig. 13b). This requires fewer bricks, but allows users to achieve a desired shape and scale anyway. This smaller tangible model can be acquired by BrickStART and scaled up to a reasonable size later on. This result can likewise be smoothed if necessary.

## 5 DISCUSSION

Building, exploring, and evaluating BrickStART has helped us understand tensions of *tangible*, *in-situ*, and *low-effort* 3D-modeling tools for personal fabrication. In the following sections, we discuss the evaluation results, followed by a broader discussion of the concept behind BrickStART.

The study confirmed that BrickStART is not only a feasible and useful tool for simple personal fabrication tasks but also one that allows users to test and iterate designs early, without fabricating design iterations [60], at the expense of fidelity or precision. This positively affected the models made by participants without any prior CAD experience, who made fewer errors regarding measurements and physics: 54.5% of the first-time 3D-modelers immediately succeeded using BrickStART, compared to 36.4% using Tinkercad, with only 9.1% of them failing to acquire the correct measures with BrickStART. In contrast, participants already familiar with 3D-modeling generally performed better with Tinkercad, as it largely follows established paradigms of solid modeling. We observed several technical issues (recognition quality and robustness), which we consider relevant technical shortcomings. Recognition of the bricks could likely be improved through other approaches that are not purely based on 2D vision [87], and further leverage the HMD as a (depth) sensor platform [82].

*Process.* A relevant disadvantage of BrickStART is the **phase disconnect** between tangible modeling and digital postprocessing. It becomes apparent when users apply postprocessing steps in the digital phases of BrickStART (e.g., scaling, or addition of primitives). While postprocessing steps are beneficial, they yield an additional disconnect from the tangible blocks: aspects like balance, clearance, or size, in general, have to be estimated visually only. This still happens in-situ, but in an intangible fashion. Postprocessing steps like smoothing had little effect on previously estimated aspects of balance and fit (e.g., fig. 9: 2, 4, 15, 16). The addition of primitives (e.g., fig. 9: 7, which, surprisingly, worked anyway) or scaling (e.g., fig. 9: 14), however, did. In contrast, Tinkercad exhibits this disconnect between the measurement phase and the modeling phase, where users made errors in dimensioning specific features of their designs (e.g., fig 9: 18, 25, 26, 32), which was identified as challenging in prior works [51].

*Tradeoffs between BrickStART and Tinkercad.* **Measurement-based errors** were more common in Tinkercad. For measurements, we observed that users generally tried to be precise in Tinkercad. This is possible and afforded by Tinkercad, but gives rise to small errors that compromise the entire results (e.g., fig. 9: 25, 32). Having to operate with the fixed grid required by BrickStART forced users to embed more clearance to interacting objects, where possible, and “*err on the side of clearance*”, which in turn does not compromise the results (e.g., fig. 9: 2, 4, 15, 16), but does not yield tight tolerances either. This contrasts fundamental notions of personal fabrication, where *precision* in manufacturing is handed to end-users [9, 22]. The absence of a readily accessible simulation component in Tinkercad made it harder to detect balance issues early on<sup>13</sup>. Users would

<sup>13</sup>After our study, Tinkercad independently added a simulation environment to the software: <https://www.tinkercad.com/blog/tinkercad-sim-lab>, Accessed 22.7.2023

have had to fabricate prototypes or set up a simulation elsewhere to detect them. This led to designs that, at first glance, seem functional, but fail when fabricated and tested (e.g., fig. 9: 19, 24, 26, 30). **Tangible testing and exploration** is at the core of BrickStART. Users could detect and compensate for some issues referring to clearance and balance (without simulation) with little to no explicit measuring effort. Simulation components are complex and, most importantly, demand even more transfers from the physical space to the digital one: not only sizes (already a source of error [51, 66] – fig. 9: 21, 26, 32) but also weight and weight distributions, for instance. In a sense, users were conducting tangible simulations within the object’s context, by building brick assemblies, testing them, and altering them if needed. Lastly, we could observe a certain straightforwardness of the participants’ use of BrickStART, as they already knew its **language of expression**. While an introduction to BrickStART or Tinkercad was needed – especially for novices – none was required for the task of “*solving a mechanical challenge with plastic bricks*”. Users immediately started to “block out” geometry and test its function using relevant objects (i.e., plant, phone). Current design tools inherently require explicit learning: their interfaces, but also their paradigms (e.g., sculpting, programmatic, parametric, or solid modeling). For toy bricks, this is not the case – expressing geometry through these bricks was natural to both novices and more experienced users in our study, in line with established benefits of tangible interaction [31, 35], and echoing insights from works employing malleable modeling materials [38, 74] or other construction kits for 3D design [5, 45]. Similarly, *testing* physical behavior in-situ with such low-fidelity models required little to no explanation – especially when comparing this to more sophisticated ways to simulate how geometry behaves (e.g., the simulation workspace in Autodesk Fusion 360). With BrickStART, we wanted users to be able to explore physical consequences early on while allowing them to **infer geometry** from 2D features. However, it is hard to holistically evaluate and explore these physical consequences with 2D outlines or their combinations: not all problems can be reliably solved in 2D or 2.5D. For instance, designing vases with a brick profile for the lathe reconstruction lets users omit the definition of the entire object. Detecting potential issues in balance or fit is then left to a purely digital preview, which has worked for some (e.g., 9: 10), but not all (e.g., 9: 14) designs made by the participants. Nevertheless, it is still possible and feasible for a subset of issues (i.e., ones that can be solved in 2D, such as the stands in figure 4, 11, and 1a). In the study, this has led to reasonably stable and fitting designs (e.g., fig. 9: 2, 4, 12). While the bricks provide a worst-case estimate of aspects like clearance, they may also induce a certain bias in the design process: constraints and approaches known from toy bricks may influence the users’ designs, such as staggered layers of bricks (e.g., 9: 4 or fig. 8: 9). Users may also think that only a rigid brick model yields a rigid fabricated (e.g., printed) model, which is incorrect. This is not necessarily the case, as a loosely assembled brick model (e.g., fig. 13a, or fig. 9: 2) still results in a solid model, while a solid brick assembly still likely indicates structural stability later on. A model that has completely disconnected parts, in turn, can be recognized as faulty early on. For novices, this was likely harder to do with Tinkercad (resulting in designs like fig. 9: 27, 29). The aforementioned aspects likely have influenced our participants’ designs (cf., section 4.1 and 4.2) and our own ones (cf., section 3 and 4.3). BrickStART can support novices in designing everyday artifacts than interact with existing counterparts while ideally reducing the potential for errors related to measuring and physics (fig. 9: 7, 16), but is constrained to a specific grid, which either leads users to embed more clearance in their designs (fig. 9: 4, 12), or attempt to correct this digitally (fig. 9: 7, 10, 11).

## 6 LIMITATIONS

*Technical Limitations.* First and foremost, the scanning process does not always yield a perfect rendition of the tangible model, especially when users try to acquire side views of complete and complex (3D) objects, instead of profiles (2D). This limits the tool with respect to the geometries

that are possible (e.g., complex cavities) and hinders insights gained with the tangible model (e.g., regarding stability), to generalize to the finished design,

*Conceptual Limitations.* BrickStARt also suffers from a certain **unidirectionality**, separating the tangible modeling phase from the remaining ones. The tangible building blocks are not synchronized with the digitally altered model. To combat this limitation, this would require approaches as presented with ReForm [93] or continuously tracked bricks, where a tangible material is kept in sync with a digital representation. This would enable bidirectionality [39] between the tangible elements and the AR design environment of BrickStARt, and support more fluid modeling processes. In its current implementation, BrickStARt is also limited with respect to object **size** and can support objects that are above a brick's size. While it is possible to use small bricks to assemble large structures, their practical benefits diminish at larger scales. Large-scale modeling (cf., [2, 41, 42]) would require a different medium, which ideally still supports tangible, in-situ design and testing.

*Evaluation.* While our evaluation results indicate a certain usefulness of BrickStARt for novices, we must consider the study's small sample size, and uneven distribution of experts vs. novices. This prohibited us from conducting meaningful statistical analyses. Subsequently, we focused on errors, error types, and their distributions, outlining trends we observed in the data and in observations. We further considered designs that failed due to technical issues with BrickStARt to be non-functional, which limited the analysis of user error (presence or absence) in these cases, despite most brick assemblies that were created hinting at a valid design. Similarly, the labeling process may not have uncovered all possible issues of a design or identified the most evident, but potentially not the most mechanically crucial issue in a model. Technical and ergonomic issues (e.g., field of view) of the headset itself also affected the ratings (e.g., SUS) and success of users.

*Impact.* The design of low-effort design tools has to be considered under the umbrella of larger-scale effects [77] of potential adoption [79], as its impact on the environment may likely be undesirable [26, 91]. Ideally, design tools for personal fabrication, operate either in reconfigurable media (e.g., bricks used by BrickStARt or comparable modules [84]), create objects that can be deconstructed [97], or operate purely digitally, to avoid the fabrication of frequent and permanent physical prototypes. We consider BrickStARt to present an approach that embraces these principles while ideally reducing [57, 58] or omitting [81] industrial notions of prototyping and iterating.

## 7 FUTURE WORK

With BrickStARt, we have presented one approach to leverage low-fidelity tangible building materials for an in-situ design process. In the future, we intend to resolve the remaining technical frictions to be able to conduct a larger-scale user study. This would also require the optimization and selection of specific interaction techniques (e.g., gestures vs. controller-based interactions) that facilitate the process outlined in Figure 2. It is furthermore intriguing to consider whether we may use *any* physical material to express designs and ideate in whichever material (or material combination) users want to or is available to them [5, 65, 78]. For future work, we are considering adding more input types to the system, such as clay [38, 74], 2D sketches [48, 73], 3D-printing pens [85] or other construction kits such as K'Nex<sup>14</sup> or Geomag<sup>15</sup>, while retaining our notion of *continuous* in-situ design. This would nudge BrickStARt further toward an approach that allows users a fully unrestricted choice of their medium of expression. Additional ways to infer geometry are conceivable for the reconstruction step, such as "Teddy" [34]. BrickStARt may also benefit from snapping: both to the initial block grid (e.g., when adding primitives later) or to the physical

<sup>14</sup><https://basicfun.com/knex/>, Accessed: 2.5.23

<sup>15</sup><https://www.geomagworld.com/en/>, Accessed: 29.4.23

environment [61]. Additional operations involving the physical environment in the design process (e.g., subtraction [82, 102]) would also add expressivity to BrickStART. Furthermore, the use of color-segmentation to extract profiles lays the groundwork for *encoding* additional information in the tangible model: users may place specific colors of bricks to encode material properties or interactive elements to be automatically generated by BrickStART (cf., [38, 74]), further decoupling the medium of expression from the medium of fabrication.

## 8 CONCLUSION

We presented BrickStART, an in-situ, tangible modeling tool consisting of a mixed-reality application combined with tangible building blocks. It enables in-situ tangible design while affording early ideation, exploration, and iteration. In our study, we demonstrated that novice users made fewer errors related to balance and fit with BrickStART, compared to Tinkercad, and presented application walkthroughs highlighting a variety of approaches enabled through BrickStART.

Starting designs for fabrication in a tangible medium is meaningful and may serve as a “kickstart” for subsequent, more digital design steps – which should remain in-situ, instead of deferring error-prone transfers to later moments of the design process (cf., [45, 51]). BrickStART enables users to tangibly explore their designs early on while enabling them to refine them meaningfully without leaving the object’s future physical context. All *relevant* process steps (i.e., ideation, design, exploration, previewing) happen in-situ, at the location of the future artifact, instead of at spatially disconnected workstations. This aspect is further supported by the lower failure rate in balance and fit exhibited by the novice users (i.e., first-time 3D-modelers) in the study. We furthermore consider BrickStART to be a step to decouple the medium of expression for the user and the medium of fabrication. By allowing users to express themselves in a highly approachable tool or material of their choice – in our case, interlocking toy bricks – users may benefit from an even lower entry barrier to 3D-design of artifacts to be fabricated, if the process is situated in the design’s future space and context.

## ACKNOWLEDGMENTS

We thank the participants of our studies for their efforts and input. We further thank the reviewers of the manuscript for their suggestions.

This research was partially funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) through the project “*Democratic and Sustainable Personal Design and Fabrication through In-situ Co-Design and Previsualization*” (project number: 525038300).

## REFERENCES

- [1] Muhammad Abdullah, Romeo Sommerfeld, Bjarne Sievers, Leonard Geier, Jonas Noack, Marcus Ding, Christoph Thieme, Laurenz Seidel, Lukas Fritzsche, Erik Langenhan, Oliver Adameck, Moritz Dzingel, Thomas Kern, Martin Taraz, Conrad Lempert, Shohei Katakura, Hany Mohsen Elhassany, Thijs Roumen, and Patrick Baudisch. 2022. HingeCore: Laser-Cut Foamcore for Fast Assembly. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology (UIST '22)*. Association for Computing Machinery, New York, NY, USA, 1–13. <https://doi.org/10.1145/3526113.3545618>
- [2] Harshit Agrawal, Udayan Umapathi, Robert Kovacs, Johannes Frohnhofen, Hsiang-Ting Chen, Stefanie Mueller, and Patrick Baudisch. 2015. Protopiper: Physically Sketching Room-Sized Objects at Actual Scale. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. ACM, New York, NY, USA, 427–436. <https://doi.org/10.1145/2807442.2807505>
- [3] Robert Aish and Peter Noakes. 1984. Architecture without Numbers – CAAD Based on a 3D Modelling System. *Computer-Aided Design* 16, 6 (Nov. 1984), 321–328. [https://doi.org/10.1016/0010-4485\(84\)90116-7](https://doi.org/10.1016/0010-4485(84)90116-7)
- [4] Celena Alcock, Nathaniel Hudson, and Parmit K. Chilana. 2016. Barriers to Using, Customizing, and Printing 3D Designs on Thingiverse. In *Proceedings of the 19th International Conference on Supporting Group Work (GROUP '16)*.



- ACM, New York, NY, USA, 195–199. <https://doi.org/10.1145/2957276.2957301>
- [5] David Anderson, James L. Frankel, Joe Marks, Aseem Agarwala, Paul Beardsley, Jessica Hodgins, Darren Leigh, Kathy Ryall, Eddie Sullivan, and Jonathan S. Yedidia. 2000. Tangible Interaction + Graphical Interpretation: A New Approach to 3D Modeling. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '00)*. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 393–402. <https://doi.org/10.1145/344779.344960>
- [6] Daniel Ashbrook, Shitao Stan Guo, and Alan Lambie. 2016. Towards Augmented Fabrication: Combining Fabricated and Existing Objects. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '16)*. ACM, New York, NY, USA, 1510–1518. <https://doi.org/10.1145/2851581.2892509>
- [7] Autodesk Inc. 2021. To Create Revolved Features | Inventor 2019 | Autodesk Knowledge Network. <https://knowledge.autodesk.com/support/inventor/learn-explore/caas/CloudHelp/cloudhelp/2019/ENU/Inventor-Help/files/GUID-401BBD2-8D5B-4300-BA8D-70037445044F-htm.html>
- [8] Autodesk Inc. 2022. Extrude Reference | Fusion 360 Help | Autodesk. <https://help.autodesk.com/view/fusion360/ENU/?guid=SLD-REF-EXTRUDE>
- [9] Patrick Baudisch and Stefanie Mueller. 2017. Personal Fabrication. *Foundations and Trends® in Human-Computer Interaction* 10, 3–4 (May 2017), 165–293. <https://doi.org/10.1561/1100000055>
- [10] Patrick Baudisch, Arthur Silber, Yannis Kommana, Milan Gruner, Ludwig Wall, Kevin Reuss, Lukas Heilman, Robert Kovacs, Daniel Rechlitz, and Thijs Roumen. 2019. Kyub: A 3D Editor for Modeling Sturdy Laser-Cut Objects. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. Association for Computing Machinery, Glasgow, Scotland Uk, 1–12. <https://doi.org/10.1145/3290605.3300796>
- [11] Alexander Berman, Francis Quek, Robert Woodward, Osazuwa Okundaye, and Jeeun Kim. 2020. 'Anyone Can Print': Supporting Collaborations with 3D Printing Services to Empower Broader Participation in Personal Fabrication. In *Proceedings of the 11th Nordic Conference on Human-Computer Interaction: Shaping Experiences, Shaping Society (NordicCHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–13. <https://doi.org/10.1145/3419249.3420068>
- [12] Srinjita Bhaduri, Quentin L Bidy, Jeffrey Bush, Abhijit Suresh, and Tamara Sumner. 2021. 3DnST: A Framework Towards Understanding Children's Interaction with Tinkercad and Enhancing Spatial Thinking Skills. In *Interaction Design and Children (IDC '21)*. Association for Computing Machinery, New York, NY, USA, 257–267. <https://doi.org/10.1145/3459990.3460717>
- [13] John Brooke. 1996. SUS-A Quick and Dirty Usability Scale. *Usability evaluation in industry* 189, 194 (1996), 4–7.
- [14] James E. Cutting and Peter M. Vishton. 1995. Perceiving Layout and Knowing Distances: The Integration, Relative Potency, and Contextual Use of Different Information about Depth. In *Perception of Space and Motion*, William Epstein and Sheena Rogers (Eds.). Academic Press, San Diego, 69–117. <https://doi.org/10.1016/B978-012240530-3/50005-5>
- [15] Mustafa Doga Dogan, Patrick Baudisch, Hrvoje Benko, Michael Nebeling, Huaishu Peng, Valkyrie Savage, and Stefanie Mueller. 2022. Fabricate It or Render It? Digital Fabrication vs. Virtual Reality for Creating Objects Instantly. In *Extended Abstracts of the 2022 CHI Conference on Human Factors in Computing Systems (CHI EA '22)*. Association for Computing Machinery, New York, NY, USA, 1–4. <https://doi.org/10.1145/3491101.3516510>
- [16] Ilker Etikan, Sulaiman Abubakar Musa, and Rukayya Sunusi Alkassim. 2016. Comparison of Convenience Sampling and Purposive Sampling. *American journal of theoretical and applied statistics* 5, 1 (2016), 1–4.
- [17] Sebastian S. Feger, Lars Semmler, Albrecht Schmidt, and Thomas Kosch. 2022. ElectronicsAR: Design and Evaluation of a Mobile and Tangible High-Fidelity Augmented Electronics Toolkit. *Proceedings of the ACM on Human-Computer Interaction* 6, ISS (Nov. 2022), 700–721. <https://doi.org/10.1145/3567740>
- [18] 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 23rd Annual ACM Symposium on User Interface Software and Technology (UIST '10)*. ACM, New York, NY, USA, 381–382. <https://doi.org/10.1145/1866218.1866230>
- [19] Sean Follmer and Hiroshi Ishii. 2012. KidCAD: Digitally Remixing Toys through Tangible Tools. In *Proceedings of the 2012 SIGCHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, 2401–2410. <https://doi.org/10.1145/2207676.2208403>
- [20] María-Elena García-Ruiz and Francisco-Javier Lena-Acebo. 2022. FabLabs: The Road to Distributed and Sustainable Technological Training through Digital Manufacturing. *Sustainability* 14, 7 (March 2022), 3938. <https://doi.org/10.3390/su14073938>
- [21] Constance Garnier and Ignasi Capdevila. 2021. Making, Hacking, Coding: Fablabs as Intermediary Platforms for Modes of Social Manufacturing. *Journal of Innovation Economics & Management* Prépublication, 0 (Dec. 2021), I–XXII. <https://doi.org/10.3917/jie.pr.1.0128>
- [22] Neil Gershenfeld. 2005. *Fab: The Coming Revolution on Your Desktop—from Personal Computers to Personal Fabrication*. Basic Books (AZ), New York, NY, USA.
- [23] Neil Gershenfeld, Alan Gershenfeld, and Joel Cutcher-Gershenfeld. 2017. *Designing Reality: How to Survive and Thrive in the Third Digital Revolution*. Basic Books, Inc., USA.

- [24] Stéphane Ginier. 2022. Triplanar - Scene | Nomad Sculpt. <https://nomadsculpt.com/manual/scene#triplanar>
- [25] Leo Gombač, Klen Čopič Pucihar, Matjaž Kljun, Paul Coulton, and Jan Grbac. 2016. 3D Virtual Tracing and Depth Perception Problem on Mobile AR. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '16)*. Association for Computing Machinery, New York, NY, USA, 1849–1856. <https://doi.org/10.1145/2851581.2892412>
- [26] Phillip Gough, Praneeth Bimsara Perera, Michael A. Kertesz, and Anusha Withana. 2023. Design, Mould, Grow!: A Fabrication Pipeline for Growing 3D Designs Using Myco-Materials. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23)*. Association for Computing Machinery, New York, NY, USA, 1–15. <https://doi.org/10.1145/3544548.3580958>
- [27] Emreca Gulay and Andrés Lucero. 2019. Integrated Workflows: Generating Feedback Between Digital and Physical Realms. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems - CHI '19*. ACM Press, Glasgow, Scotland Uk, 1–15. <https://doi.org/10.1145/3290605.3300290>
- [28] Sandra G. Hart. 2006. NASA-task Load Index (NASA-TLX); 20 Years Later. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, Vol. 50. Sage Publications Sage CA: Los Angeles, CA, Los Angeles, California, USA, 904–908.
- [29] Florian Heller, Jan Thar, Dennis Lewandowski, Mirko Hartmann, Pierre Schoonbrood, Sophy Stöner, Simon Voelker, and Jan Borchers. 2018. CutCAD - An Open-source Tool to Design 3D Objects in 2D. In *Proceedings of the 2018 Designing Interactive Systems Conference (DIS '18)*. Association for Computing Machinery, New York, NY, USA, 1135–1139. <https://doi.org/10.1145/3196709.3196800>
- [30] Gary T. Henry. 1990. *Practical Sampling*. Applied Social Research Methods, Vol. 21. SAGE Publications, Inc., London, UK.
- [31] Eva Hornecker and Jacob Buur. 2006. Getting a Grip on Tangible Interaction: A Framework on Physical Space and Social Interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '06)*. Association for Computing Machinery, New York, NY, USA, 437–446. <https://doi.org/10.1145/1124772.1124838>
- [32] Yi-Jheng Huang, Kang-Yi Liu, Suiang-Shyan Lee, and I-Cheng Yeh. 2021. Evaluation of a Hybrid of Hand Gesture and Controller Inputs in Virtual Reality. *International Journal of Human-Computer Interaction* 37, 2 (Jan. 2021), 169–180. <https://doi.org/10.1080/10447318.2020.1809248>
- [33] 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)*. ACM, New York, NY, USA, 384–396. <https://doi.org/10.1145/2858036.2858266>
- [34] Takeo Igarashi, Satoshi Matsuoka, and Hidehiko Tanaka. 1999. Teddy: A Sketching Interface for 3D Freeform Design. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '99)*. ACM Press/Addison-Wesley Publishing Co., USA, 409–416. <https://doi.org/10.1145/311535.311602>
- [35] Hiroshi Ishii. 2008. Tangible Bits: Beyond Pixels. In *Proceedings of the 2Nd International Conference on Tangible and Embedded Interaction (TEI '08)*. ACM, New York, NY, USA, xv–xxv. <https://doi.org/10.1145/1347390.1347392>
- [36] Robert J.K. Jacob, Audrey Girouard, Leanne M. Hirshfield, Michael S. Horn, Orit Shaer, Erin Treacy Solovey, and Jamie Zigelbaum. 2008. Reality-Based Interaction: A Framework for Post-WIMP Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '08)*. Association for Computing Machinery, Florence, Italy, 201–210. <https://doi.org/10.1145/1357054.1357089>
- [37] Yunwoo Jeong, Hyungjun Cho, Taewan Kim, and Tek-Jin Nam. 2023. AutomataStage: An AR-mediated Creativity Support Tool for Hands-on Multidisciplinary Learning. 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/10.1145/3544548.3581408>
- [38] Michael D. Jones, Kevin Seppi, and Dan R. Olsen. 2016. What You Sculpt Is What You Get: Modeling Physical Interactive Devices with Clay and 3D Printed Widgets. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 876–886. <https://doi.org/10.1145/2858036.2858493>
- [39] Hiroki Kaimoto, Kyzyl Monteiro, Mehrad Faridan, Jiatong Li, Samin Farajian, Yasuaki Kakehi, Ken Nakagaki, and Ryo Suzuki. 2022. Sketched Reality: Sketching Bi-Directional Interactions Between Virtual and Physical Worlds with AR and Actuated Tangible UI. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology (UIST '22)*. Association for Computing Machinery, New York, NY, USA, 1–12. <https://doi.org/10.1145/3526113.3545626>
- [40] Jeeun Kim, Anhong Guo, Tom Yeh, Scott E. Hudson, and Jennifer Mankoff. 2017. Understanding Uncertainty in Measurement and Accommodating Its Impact in 3D Modeling and Printing. In *Proceedings of the 2017 Conference on Designing Interactive Systems (DIS '17)*. ACM, New York, NY, USA, 1067–1078. <https://doi.org/10.1145/3064663.3064690>
- [41] Robert Kovacs, Lukas Rambold, Lukas Fritzsche, Dominik Meier, Jotaro Shigeyama, Shohei Katakura, Ran Zhang, and Patrick Baudisch. 2021. Trusscillator: A System for Fabricating Human-Scale Human-Powered Oscillating Devices. In *The 34th Annual ACM Symposium on User Interface Software and Technology*. Association for Computing Machinery, New York, NY, USA, 1074–1088. <https://doi.org/10.1145/3472749.3474807>

- [42] Robert Kovacs, Anna Seufert, Ludwig Wall, Hsiang-Ting Chen, Florian Meinel, Willi Müller, Sijing You, Maximilian Brehm, Jonathan Striebel, Yannis Kommana, Alexander Popiak, Thomas Bläsius, and Patrick Baudisch. 2017. TrussFab: Fabricating Sturdy Large-Scale Structures on Desktop 3D Printers. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. Association for Computing Machinery, Denver, Colorado, USA, 2606–2616. <https://doi.org/10.1145/3025453.3026016>
- [43] Ben Lafreniere and Tovi Grossman. 2018. Blocks-to-CAD: A Cross-Application Bridge from Minecraft to 3D Modeling. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology (UIST '18)*. Association for Computing Machinery, New York, NY, USA, 637–648. <https://doi.org/10.1145/3242587.3242602>
- [44] Manfred Lau, Greg Saul, Jun Mitani, and Takeo Igarashi. 2010. Modeling-in-Context: User Design of Complementary Objects with a Single Photo. In *Proceedings of the Seventh Sketch-Based Interfaces and Modeling Symposium (SBIM '10)*. Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 17–24. <http://dl.acm.org/citation.cfm?id=1923363.1923367>
- [45] Danny Leen, Raf Ramakers, and Kris Luyten. 2017. StrutModeling: A Low-Fidelity Construction Kit to Iteratively Model, Test, and Adapt 3D Objects. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST '17)*. ACM, New York, NY, USA, 471–479. <https://doi.org/10.1145/3126594.3126643>
- [46] Maximilian Letter and Katrin Wolf. 2022. Tangible Version Control: Exploring a Physical Object's Alternative Versions. In *Extended Abstracts of the 2022 CHI Conference on Human Factors in Computing Systems (CHI EA '22)*. Association for Computing Machinery, New York, NY, USA, 1–7. <https://doi.org/10.1145/3491101.3519686>
- [47] James R. Lewis. 1992. Psychometric Evaluation of the Post-Study System Usability Questionnaire: The PSSUQ. In *Proceedings of the Human Factors Society Annual Meeting*, Vol. 36. Sage Publications Sage CA: Los Angeles, CA, Washington, DC, USA, 1259–1260.
- [48] Changjian Li, Hao Pan, Adrien Bousseau, and Niloy J. Mitra. 2020. Sketch2CAD: Sequential CAD Modeling by Sketching in Context. *ACM Transactions on Graphics* 39, 6 (Nov. 2020), 164:1–164:14. <https://doi.org/10.1145/3414685.3417807>
- [49] Chen Liang, Anhong Guo, and Jeeun Kim. 2022. CustomizAR: Facilitating Interactive Exploration and Measurement of Adaptive 3D Designs. In *Designing Interactive Systems Conference (DIS '22)*. Association for Computing Machinery, New York, NY, USA, 898–912. <https://doi.org/10.1145/3532106.3533561>
- [50] Can Liu, Chenyue Dai, Qingzhou Ma, Brinda Mehra, and Alvaro Cassinelli. 2022. AngleCAD: Surface-Based 3D Modelling Techniques on Foldable Touchscreens. *Proceedings of the ACM on Human-Computer Interaction* 6, ISS (Nov. 2022), 582:592–582:616. <https://doi.org/10.1145/3567735>
- [51] Chandan Mahapatra, Jonas Kjeldmand Jensen, Michael McQuaid, and Daniel Ashbrook. 2019. Barriers to End-User Designers of Augmented Fabrication. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. ACM, New York, NY, USA, 383:1–383:15. <https://doi.org/10.1145/3290605.3300613>
- [52] James McCrae, Nobuyuki Umetani, and Karan Singh. 2014. FlatFitFab: Interactive Modeling with Planar Sections. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology (UIST '14)*. ACM, New York, NY, USA, 13–22. <https://doi.org/10.1145/2642918.2647388>
- [53] Daniela Mitterberger. 2022. Augmented Human, Extended Machine: Extended Reality Systems for Robotic Fabrication in Architecture, Engineering, and Construction. *XRDS: Crossroads, The ACM Magazine for Students* 29, 1 (Oct. 2022), 48–53. <https://doi.org/10.1145/3558196>
- [54] Daniela Mitterberger, Evgenia-Makrina Angelaki, Foteini Salveridou, Romana Rust, Lauren Vasey, Fabio Gramazio, and Matthias Kohler. 2023. Extended Reality Collaboration: Virtual and Mixed Reality System for Collaborative Design and Holographic-Assisted On-site Fabrication. In *Towards Radical Regeneration*, Christoph Gengnagel, Olivier Baverel, Giovanni Betti, Mariana Popescu, Mette Ramsgaard Thomsen, and Jan Wurm (Eds.). Springer International Publishing, Cham, 283–295. [https://doi.org/10.1007/978-3-031-13249-0\\_24](https://doi.org/10.1007/978-3-031-13249-0_24)
- [55] Daniela Mitterberger, Kathrin Dörfler, Timothy Sandy, Foteini Salveridou, Marco Hutter, Fabio Gramazio, and Matthias Kohler. 2020. Augmented Bricklaying. *Construction Robotics* 4, 3 (Dec. 2020), 151–161. <https://doi.org/10.1007/s41693-020-00035-8>
- [56] Stefanie Mueller. 2017. 3D Printing for Human-computer Interaction. *Interactions* 24, 5 (Aug. 2017), 76–79. <https://doi.org/10.1145/3125399>
- [57] Stefanie Mueller. 2018. Toward Direct Manipulation for Personal Fabrication. *IEEE Pervasive Computing* 17, 1 (Jan. 2018), 75–81. <https://doi.org/10.1109/MPRV.2018.011591064>
- [58] Stefanie Mueller, Sangha Im, Serafima Gurevich, Alexander Teibrich, Lisa Pfisterer, François Guimbretière, and Patrick Baudisch. 2014. WirePrint: 3D Printed Previews for Fast Prototyping. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology (UIST '14)*. ACM, New York, NY, USA, 273–280. <https://doi.org/10.1145/2642918.2647359>
- [59] Stefanie Mueller, Bastian Kruck, and Patrick Baudisch. 2013. LaserOrigami: Laser-Cutting 3D Objects. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York,

- NY, USA, 2585–2592. <https://doi.org/10.1145/2470654.2481358>
- [60] Stefanie Mueller, Tobias Mohr, Kerstin Guenther, Johannes Frohnhofen, and Patrick Baudisch. 2014. faBrickation: Fast 3D Printing of Functional Objects by Integrating Construction Kit Building Blocks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 3827–3834. <https://doi.org/10.1145/2556288.2557005>
- [61] Benjamin Nuernberger, Eyal Ofek, Hrvoje Benko, and Andrew D. Wilson. 2016. SnapToReality: Aligning Augmented Reality to the Real World. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 1233–1244. <https://doi.org/10.1145/2858036.2858250>
- [62] Keunwoo Park, Conrad Lempert, Muhammad Abdullah, Shohei Katakura, Jotaro Shigeyama, Thijs Roumen, and Patrick Baudisch. 2022. FoolProofJoint: Reducing Assembly Errors of Laser Cut 3D Models by Means of Custom Joint Patterns. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22)*. Association for Computing Machinery, New York, NY, USA, 1–12. <https://doi.org/10.1145/3491102.3501919>
- [63] Huaishu Peng, Jimmy Briggs, Cheng-Yao Wang, Kevin Guo, Joseph Kider, Stefanie Mueller, Patrick Baudisch, and François Guimbretière. 2018. RoMA: Interactive Fabrication with Augmented Reality and a Robotic 3D Printer. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, 579:1–579:12. <https://doi.org/10.1145/3173574.3174153>
- [64] Pixologic Inc. 2021. ShadowBox | ZBrush Docs. <http://docs.pixologic.com/user-guide/3d-modeling/modeling-basics/creating-meshes/shadowbox/>
- [65] Cecil Piya, Vinayak, Yunbo Zhang, and Karthik Ramani. 2016. RealFusion: An Interactive Workflow for Repurposing Real-World Objects towards Early-stage Creative Ideation. In *Graphics Interface*, Vol. 42. Canadian Human-Computer Communications Society - Société canadienne du dialogue humain-machine, Victoria, British Columbia, Canada, 8. <https://doi.org/10.20380/GI2016.11>
- [66] Raf Ramakers, Danny Leen, Jeeun Kim, Kris Luyten, Steven Houben, and Tom Veuskens. 2023. Measurement Patterns: User-Oriented Strategies for Dealing with Measurements and Dimensions in Making Processes. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23)*. Association for Computing Machinery, New York, NY, USA, 1–17. <https://doi.org/10.1145/3544548.3581157>
- [67] Thierry Rayna and Ludmila Striukova. 2016. From Rapid Prototyping to Home Fabrication: How 3D Printing Is Changing Business Model Innovation. *Technological Forecasting and Social Change* 102 (Jan. 2016), 214–224. <https://doi.org/10.1016/j.techfore.2015.07.023>
- [68] Patrick Reipschläger and Raimund Dachselt. 2019. DesignAR: Immersive 3D-Modeling Combining Augmented Reality with Interactive Displays. In *Proceedings of the 2019 ACM International Conference on Interactive Surfaces and Spaces (ISS '19)*. Association for Computing Machinery, Daejeon, Republic of Korea, 29–41. <https://doi.org/10.1145/3343055.3359718>
- [69] Mitchel Resnick, John Maloney, Andrés Monroy-Hernández, Natalie Rusk, Evelyn Eastmond, Karen Brennan, Amon Millner, Eric Rosenbaum, Jay Silver, Brian Silverman, and Yasmin Kafai. 2009. Scratch: Programming for All. *Commun. ACM* 52, 11 (Nov. 2009), 60–67. <https://doi.org/10.1145/1592761.1592779>
- [70] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. 2004. "GrabCut": Interactive Foreground Extraction Using Iterated Graph Cuts. In *ACM SIGGRAPH 2004 Papers (SIGGRAPH '04)*. Association for Computing Machinery, New York, NY, USA, 309–314. <https://doi.org/10.1145/1186562.1015720>
- [71] Thijs Jan Roumen, Willi Müller, and Patrick Baudisch. 2018. Grafter: Remixing 3D-Printed Machines. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, 63:1–63:12. <https://doi.org/10.1145/3173574.3173637>
- [72] Calvin Rubens, Sean Braley, Julie Torpegaard, Nicklas Lind, Roel Vertegaal, and Timothy Merritt. 2020. Flying LEGO Bricks: Observations of Children Constructing and Playing with Programmable Matter. In *Proceedings of the Fourteenth International Conference on Tangible, Embedded, and Embodied Interaction*. ACM, Sydney NSW Australia, 193–205. <https://doi.org/10.1145/3374920.3374948>
- [73] Greg Saul, Manfred Lau, Jun Mitani, and Takeo Igarashi. 2011. SketchChair: An All-in-one Chair Design System for End Users. In *Proceedings of the Fifth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '11)*. ACM, New York, NY, USA, 73–80. <https://doi.org/10.1145/1935701.1935717>
- [74] 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)*. ACM, New York, NY, USA, 103–108. <https://doi.org/10.1145/2807442.2807508>
- [75] Stefan Schneeeggass, Alireza Sahami Shirazi, Tanja Döring, David Schmid, and Albrecht Schmidt. 2014. NatCut: An Interactive Tangible Editor for Physical Object Fabrication. In *Proceedings of the Extended Abstracts of the 32Nd Annual ACM Conference on Human Factors in Computing Systems (CHI EA '14)*. ACM, New York, NY, USA, 1441–1446. <https://doi.org/10.1145/2559206.2581189>



- [76] Rita Shewbridge, Amy Hurst, and Shaun K. Kane. 2014. Everyday Making: Identifying Future Uses for 3D Printing in the Home. In *Proceedings of the 2014 Conference on Designing Interactive Systems (DIS '14)*. ACM, New York, NY, USA, 815–824. <https://doi.org/10.1145/2598510.2598544>
- [77] Ben Shneiderman and Anne Rose. 1996. Social Impact Statements: Engaging Public Participation in Information Technology Design. In *Proceedings of the Symposium on Computers and the Quality of Life (CQL '96)*. Association for Computing Machinery, New York, NY, USA, 90–96. <https://doi.org/10.1145/238339.238378>
- [78] Evgeny Stemasov. 2021. Enabling Ubiquitous Personal Fabrication by Deconstructing Established Notions of Artifact Modeling. In *The Adjunct Publication of the 34th Annual ACM Symposium on User Interface Software and Technology (UIST '21)*. Association for Computing Machinery, New York, NY, USA, 166–170. <https://doi.org/10.1145/3474349.3477589>
- [79] Evgeny Stemasov, Alexander Botner, Enrico Rukzio, and Jan Gugenheimer. 2022. Ephemeral Fabrication: Exploring a Ubiquitous Fabrication Scenario of Low-Effort, In-Situ Creation of Short-Lived Physical Artifacts. In *Sixteenth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '22)*. ACM, New York, NY, USA, 1–17. <https://doi.org/10.1145/3490149.3501331>
- [80] Evgeny Stemasov, David Ledo, George Fitzmaurice, and Fraser Anderson. 2023. Immersive Sampling: Exploring Sampling for Future Creative Practices in Media-Rich, Immersive Spaces. In *Proceedings of the 2023 ACM Designing Interactive Systems Conference (DIS '23)*. Association for Computing Machinery, New York, NY, USA, 212–229. <https://doi.org/10.1145/3563657.3596131>
- [81] Evgeny Stemasov, Enrico Rukzio, and Jan Gugenheimer. 2021. The Road to Ubiquitous Personal Fabrication: Modeling-Free Instead of Increasingly Simple. *IEEE Pervasive Computing* 20, 1 (2021), 1–9. <https://doi.org/10.1109/MPRV.2020.3029650>
- [82] Evgeny Stemasov, Tobias Wagner, Jan Gugenheimer, and Enrico Rukzio. 2020. Mix&Match: Towards Omitting Modelling Through In-situ Remixing of Model Repository Artifacts in Mixed Reality. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. ACM, New York, NY, USA, 1–12. <https://doi.org/10.1145/3313831.3376839>
- [83] Evgeny Stemasov, Tobias Wagner, Jan Gugenheimer, and Enrico Rukzio. 2022. ShapeFindAR: Exploring In-Situ Spatial Search for Physical Artifact Retrieval Using Mixed Reality. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22)*. Association for Computing Machinery, New York, NY, USA, 1–12. <https://doi.org/10.1145/3491102.3517682>
- [84] Ryo Suzuki, Junichi Yamaoka, Daniel Leithinger, Tom Yeh, Mark D. Gross, Yoshihiro Kawahara, and Yasuaki Kakehi. 2018. Dynablock: Dynamic 3D Printing for Instant and Reconstructable Shape Formation. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology (UIST '18)*. ACM, New York, NY, USA, 99–111. <https://doi.org/10.1145/3242587.3242659>
- [85] Haruki Takahashi and Jeeun Kim. 2019. 3D Pen + 3D Printer: Exploring the Role of Humans and Fabrication Machines in Creative Making. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. ACM, New York, NY, USA, 295:1–295:12. <https://doi.org/10.1145/3290605.3300525>
- [86] Keiichiro Taniguchi and Tomoko Hashida. 2018. Rapid Prototyping System Using Transformable and Adherable PCL Blocks. In *SIGGRAPH Asia 2018 Posters (SA '18)*. ACM, New York, NY, USA, 50:1–50:2. <https://doi.org/10.1145/3283289.3283308>
- [87] George Tattersall, Dizhong Zhu, William A. P. Smith, Sebastian Deterding, and Patrik Huber. 2021. Reconstructing Creative Lego Models. In *Computer Vision – ACCV 2020*, Hiroshi Ishikawa, Cheng-Lin Liu, Tomas Pajdla, and Jianbo Shi (Eds.), Vol. 12622. Springer International Publishing, Cham, 55–70. [https://doi.org/10.1007/978-3-030-69525-5\\_4](https://doi.org/10.1007/978-3-030-69525-5_4)
- [88] Udayan Umapathi, Hsiang-Ting Chen, Stefanie Mueller, Ludwig Wall, Anna Seufert, and Patrick Baudisch. 2015. LaserStacker: Fabricating 3D Objects by Laser Cutting and Welding. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. ACM, New York, NY, USA, 575–582. <https://doi.org/10.1145/2807442.2807512>
- [89] Tom Valkeneers, Danny Leen, Daniel Ashbrook, and Raf Ramakers. 2019. StackMold: Rapid Prototyping of Functional Multi-Material Objects with Selective Levels of Surface Details. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology (UIST '19)*. Association for Computing Machinery, New York, NY, USA, 687–699. <https://doi.org/10.1145/3332165.3347915>
- [90] Russell Wade, Nigel P. Garland, and Gary Underwood. 2017. Challenges of 3D Printing for Home Users. In *19th International Conference on Engineering and Product Design Education*, A. Berg, E. Bohemia, L. Buck, T. Gulden, A. Kovacevic, and N. Pavel (Eds.). Design Society and IED, Oslo, Norway, 478–484. <http://eprints.bournemouth.ac.uk/29947/>
- [91] Ludwig Wilhelm Wall, Alec Jacobson, Daniel Vogel, and Oliver Schneider. 2021. Scrapy: Using Scrap Material as Infill to Make Fabrication More Sustainable. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 1–12. <https://doi.org/10.1145/3411764.3445187>



- [92] Christian Weichel, Jason Alexander, Abhijit Karnik, and Hans Gellersen. 2015. SPATA: Spatio-Tangible Tools for Fabrication-Aware Design. In *Proceedings of the Ninth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '15)*. ACM, New York, NY, USA, 189–196. <https://doi.org/10.1145/2677199.2680576>
- [93] 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)*. ACM, New York, NY, USA, 93–102. <https://doi.org/10.1145/2807442.2807451>
- [94] 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)*. ACM, New York, NY, USA, 3855–3864. <https://doi.org/10.1145/2556288.2557090>
- [95] Michael Wessely, Theophanis Tsandilas, and Wendy E. Mackay. 2018. Shape-Aware Material: Interactive Fabrication with ShapeMe. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology (UIST '18)*. ACM, New York, NY, USA, 127–139. <https://doi.org/10.1145/3242587.3242619>
- [96] Wikibooks. 2022. 2D to 3D Extrusion - OpenSCAD User Manual. [https://en.wikibooks.org/wiki/OpenSCAD\\_User\\_Manual/2D\\_to\\_3D\\_Extrusion](https://en.wikibooks.org/wiki/OpenSCAD_User_Manual/2D_to_3D_Extrusion)
- [97] Shanel Wu and Laura Devendorf. 2020. Unfabricate: Designing Smart Textiles for Disassembly. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–14. <https://doi.org/10.1145/3313831.3376227>
- [98] Tom Yeh and Jeeun Kim. 2018. CraftML: 3D Modeling Is Web Programming. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. Association for Computing Machinery, Montreal QC, Canada, 1–12. <https://doi.org/10.1145/3173574.3174101>
- [99] Nur Yildirim, James McCann, and John Zimmerman. 2020. Digital Fabrication Tools at Work: Probing Professionals' Current Needs and Desired Futures. 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/10.1145/3313831.3376621>
- [100] Derek M. Yip-Hoi and Jeffrey L. Newcomer. 2011. Teaching CAD Modeling Using LEGO. In *2011 ASEE Annual Conference & Exposition*. American Society for Engineering Education, Washington DC, 22.1374.1–22.1374.12. <https://peer.asee.org/teaching-cad-modeling-using-lego>
- [101] Ya-Ting Yue, Xiaolong Zhang, Yongliang Yang, Gang Ren, Yi-King Choi, and Wenping Wang. 2017. WireDraw: 3D Wire Sculpturing Guided with Mixed Reality. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 3693–3704. <https://doi.org/10.1145/3025453.3025792>
- [102] Kening Zhu, Alexandru Dancu, and Shengdong (Shen) Zhao. 2016. FusePrint: A DIY 2.5D Printing Technique Embracing Everyday Artifacts. In *Proceedings of the 2016 ACM Conference on Designing Interactive Systems (DIS '16)*. ACM, New York, NY, USA, 146–157. <https://doi.org/10.1145/2901790.2901792>

Received 2023-02-15; accepted 2023-05-30