

Making despite Material Constraints with Augmented Reality-Mediated Prototyping

Sowmya Somanath, Lora Oehlberg, Ehud Sharlin
University of Calgary
Calgary, Canada,
{ssomanat,lora.oehlberg,ehud}@ucalgary.ca

ABSTRACT

Makers build physical interactive objects using programmable electronics. However, makers often must stop building when material resources (e.g., electronic components) are not immediately or easily available. In this paper, we envision Augmented Reality (AR)-mediated prototyping as a way for makers to continue building physical computing projects despite a lack of electronic components. *AR-mediated prototyping* enables makers to build, program, interact with, and iterate on physical computing projects that combine both real and stand-in virtual electronic components. We designed and implemented a technology probe, *Polymorphic Cube* (PMC), as an instantiation of our vision. We introduced PMC to twelve makers and asked them to build four simple lamp prototypes despite missing I/O components (push button, LED, light sensor, servo). Our results show that PMC helped participants prototype despite missing components, and highlighted how AR-mediated prototyping extends to exploring project ideas, tinkering with implementation, and making with others.

ACM Classification Keywords

H.5.m. Information Interfaces and Presentation (e.g. HCI): Miscellaneous

Author Keywords

Material Constraints; Making; Physical Computing; Prototyping

INTRODUCTION

As part of the Maker Movement, people build physical computing projects such as toys, robots, and utilitarian products using art and craft materials, programmable electronics, software, and fabrication tools. The vision of the Maker Movement is to democratize technological practices and transform people into producers of artifacts and knowledge [28].

However, the Maker Movement is only available to those with access to materials for making [6]. Material cost or accessibility concerns present potential roadblocks to building physical computing projects [25, 26]. When electronic components are not available, makers may be forced to conduct iterative

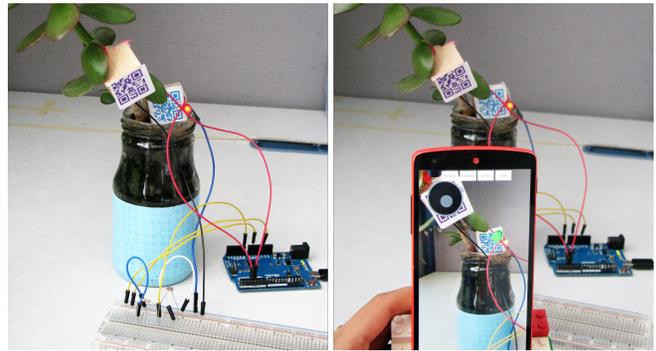


Figure 1. AR-Mediated Prototyping blends real and virtual components to create physical computing projects despite missing materials. Above, a plant monitoring system prototyped using our technology probe (Polymorphic Cube).

on-line or empirical research to find alternatives [25]. However, not all makers have independent learning resources (e.g., the Internet) to conduct such research [26]. Some makers may become discouraged and discard their original project ideas entirely [26]. This inequity compromises the democratic vision of the Maker Movement; by helping people continue to make despite missing material resources, we can equitably extend the reach of the Maker Movement’s vision.

One possible response to missing materials is to digitally create and simulate physical circuits. Researchers have proposed electronics simulation softwares (e.g., Circuits.IO [3]), using which makers can virtually explore “what-if” scenarios easily and instantly when no electronics are available. However, this undermines the essence of building physical computing projects – physical computing projects embrace the physicality of electronic components, circuitry, and interactions [22]. We suggest pursuing an intermediary between purely virtual and purely physical representations of the project (Figure 1). *How might we blend a virtual simulation of missing components with real-world physical prototyping materials to address a lack of materials?*

Augmented Reality (AR) is one way to blend virtual simulation and physical prototyping materials by superimposing computer-generated digital content on a real world object [20, 30]. In this paper, we propose *AR-mediated prototyping* – an approach to prototyping physical computing projects with both real and virtual electronic components (Figure 1). AR-mediated prototyping allows makers to leverage AR to continue to build as much as possible until missing electronic components become available.

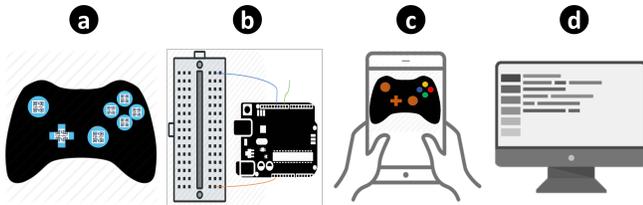


Figure 2. Parts of technology for AR-Mediated Prototyping: (a) a trackable game placeholder object, (b) programmable microcontroller and circuit, (c) companion AR application, and (d) programming IDE.

In the remainder of this paper, we first present our vision for designing technology for *AR-mediated prototyping*. Next, we discuss the design and implementation of a technology probe, *Polymorphic Cube* (PMC) based on our vision. Then, we describe two studies that capture makers’ reactions to our technology probe and the broader vision of AR-mediated prototyping. Finally, we conclude with lessons learned about how AR-mediated prototyping allows physical computing projects to continue despite material constraints.

AR-MEDIATED PROTOTYPING: VISION

AR-mediated prototyping lets makers substitute virtual stand-in components for missing electronics. Makers connect components to, interact with, and program a unified circuit that includes both real-world and virtual materials (Figure 1).

Our vision for designing technology for AR-mediated prototyping consists of four parts (Figure 2): (a) a trackable physical placeholder object (e.g., a game controller-like object as shown in Figure 2a), (b) a programmable microcontroller (e.g., Arduino as shown in Figure 2b), (c) a companion AR application (Figure 2c), and (d) a programming IDE (Figure 2d). To build circuits the maker connects the physical placeholder object to a microcontroller pin. The maker uses the companion AR application to assign the placeholder object to a selected missing electronic component (e.g., LED, servo, photocell). Makers can program both connected AR components and real-world components in the same microcontroller-specific programming language.

Within this base vision for AR-mediated prototyping, there are several aspects to consider: (1) physical and virtual form of AR components, (2) circuitry and programming, (3) physical interactions, and (4) social interactions. In this section, we discuss these aspects and suggest four goals for designing technology for AR-mediated prototyping. We also present three possible high-level tasks that AR-mediated prototyping can support based on the four aspects and the goals for technology.

Physical and Virtual Form

Unlike real components, AR components are linked in both physical and virtual space. In our vision, makers choose the dimensions and physical appearance of the placeholder object; this could be a physical replica of the missing component, any readily-available found object, or even an enclosure for the final circuit. For example, Figure 2a shows the use of a game controller-like object as a placeholder object. Because the placeholder object is maker-defined, the AR tags that link physical and virtual representations need to accommodate

different shapes and sizes, and readily attach to a variety of placeholder materials. For example, Figure 2a shows the use of circular and cross shaped AR tags.

We envision that makers assign a placeholder object to a virtual electronic component, represented in virtual space by a 3D model. The AR component may represent a single electronic component, or a more complex subcircuit. The AR component can represent components that are either analog or digital, and either input or output. AR components should also simulate physical behaviors of electronic components, such rotation speed of a motor, or motion of actuated sliders via appropriate animations.

Based on these considerations, we suggest that the technology for AR-mediated prototyping should help makers easily build (construct and assign) AR components (*Goal #1*).

Circuitry and Programming

Physical computing projects require both physical circuit building and microcontroller programming. We envision that AR-mediated prototyping should build on skills that makers have learned from using existing physical computing kits. For example, while some physical computing kits require traditional circuit-building skills using breadboard and wires (e.g., Arduino), others are more plug-and-play (e.g., Phidgets, LittleBits). We envision that makers will connect AR components to the circuit in the same manner as real-world components connect to the maker’s preferred microcontroller platform. The microcontroller then detects both the real and the AR components that are connected to the circuit.

Makers program microcontrollers to build interactivity into physical computing projects. We envision that makers will program AR components in the same programming language and code base that defines the behavior of real-world components. This allows the hybrid ecosystem of real and virtual components to behave as a single system.

Based on these considerations, technology for AR-mediated prototyping should enable coupling – via circuitry and programming – the AR components and real-world prototyping materials to create a unified project (*Goal #2*).

Physical Interactions

In physical computing projects, people physically interact with the electronic components. In the case of AR-mediated prototyping, people need to simultaneously interact with AR and real-world components. Possible ways of interacting with AR components will depend on the AR technology being used.

For example, for AR-mediated prototyping using mobile devices such as smartphones or tablets, we identified four possible interaction styles (Figure 3), based on a continuum of AR interaction paradigms [13]. First, the maker could use an *on-screen widget* (Figure 3a), e.g., controlling the direction and position of a servomotor using an on-screen widget. Second, the maker could use *touch interactions* (Figure 3b), e.g., pressing a button by tapping its virtual representation on a capacitive touchscreen. Third, a maker could interact with *built in phone sensors* (Figure 3c), e.g., changing the volume of a virtual mini speaker by using the volume controller buttons

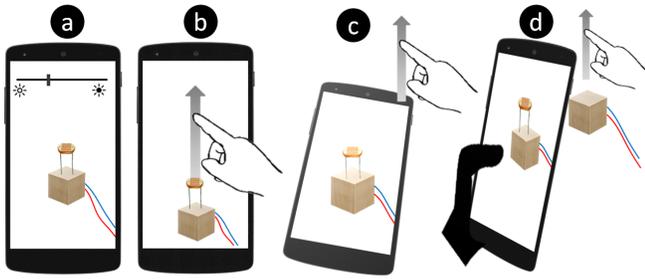


Figure 3. Interacting with AR light sensor component using (a) on-screen widget, (b) touch, (c) built in phone sensors, and (d) physical placeholder object.

on the side of the phone. Fourth, a maker could interact with the *physical placeholder object* (Figure 3d), e.g., controlling a rotation sensor by physically rotating the placeholder object. Alternative AR platforms (e.g., HoloLens) may be able to use some of these interaction styles, but may also introduce additional ways of interacting with AR components. Based on this consideration, technology for AR-mediated prototyping should support appropriate interaction with AR components (*Goal #3*).

Social Interactions

Materials also play a social role in online and real-world makerspaces, as makers share electronic components, as well as its corresponding software code or project documentation [9]. Traditionally, people share physical electronic components separately from software code. However, with AR-mediated prototyping we envision that makers can assign an AR component and its corresponding software code to a single AR-tag. Makers can share the AR tag physically in co-located settings, or post online for others to download. Based on this consideration, technology for AR-mediated prototyping should support sharing both components and code as a single unit (*Goal #4*).

Possible Tasks

We envision people might use the AR-mediated prototyping tool for three types of tasks. These tasks serve as a starting point to explore the different aspects of building a physical computing project, and are not meant to be an exhaustive list of possibilities. In this section, we present our suggested list of possible tasks and a walk-through of the steps required to accomplish the tasks.

Task #1. Experiment with a variety of components: To test design ideas, makers iteratively assign an AR component to different electronic components. For this task the maker first constructs an AR component by attaching an AR tag to a physical placeholder object, and then, assigns the AR component to electronic component(s) using the companion application. This task can be primarily supported by technology that allows makers to easily build AR components (*Goal #1*).

Task #2. Prepare for transitioning to real-world components: To build a high-fidelity system makers switch the AR components to real-world components. This could happen after a prototype idea is finalized and/or when the components become available. As part of this task, first the maker builds a physical circuit by connecting both the real-world and AR

electronic components. Second, the maker writes code to program the circuit. Third, the maker tests the circuit by interacting with the electronic components. Lastly, maker swaps the AR components with the real-world components. This task can be primarily supported by technology that allows makers to couple AR components and real-world prototyping materials (*Goal #2*), and test the circuit by interacting with the components (*Goal #3*).

Task #3. Share components and code: To work on projects in a social setting, makers collaborate with others and share electronic components and software code. As part of this task, the maker saves the electronic component(s) and software code in a specific AR tag. In co-located settings the maker then hands out this AR tag to others. For remote sharing, the maker uploads the AR tag and shares a link to download. This task be supported by technology that enables sharing of AR tags mapped to specific code and components (*Goal #4*).

RELATED LITERATURE

AR-mediated prototyping approach to making within material constraints is a new idea. Polymorphic Cube (PMC), a technology probe for prototyping physical circuits using real-world and virtual electronic components, represents a first exploration of an AR-mediated prototyping tool. Here we discuss previous research in authoring environments, tabletop systems, and augmented reality that has influenced the design of PMC.

Authoring Environment and Augmented Workspaces

Authoring environments (e.g., [15]) and augmented workspaces (e.g., [12, 29]) can help bridge the gap between the physical and the digital worlds. d.tools [15] is an integrated authoring environment for designing, analyzing, and testing early prototypes of physical user interfaces. Designers begin by plugging physical components into the d.tools hardware interface and then author interaction behavior digitally using a visual programming interface. The learner triggers the interaction model by either interacting with the physical electronics or by simulating the virtual version of the plugged electronic component. The goal of d.tools is to support design thinking by simplifying the implementation phase.

Conradi et al. [12] presented the Flow of Electrons, a tabletop system to help users learn about electronics. Users place various tagged electronics such as microcontrollers and electronic components on the tabletop, and using touch, virtually build the circuit. Animations of digital wires confirm correct wiring – as soon as the virtual circuit is closed, animated “electrons” start to flow. react3D [29] is also a tabletop system used to explore circuit building. react3D [29] uses abstract tangibles to serve as electronic components. One tangible block is connected to a power supply to power the circuits. However, to test interactions (e.g., turning an LED on/off) real electronic components (e.g., LED) must be attached to the representative tangible block.

Our PMC prototype is inspired by d.tools [15] and the tabletop systems [12, 29], which demonstrate the ability to start exploring physical circuits without necessarily having all the required components. However, our goal is different from these systems – we want to help makers continue to build

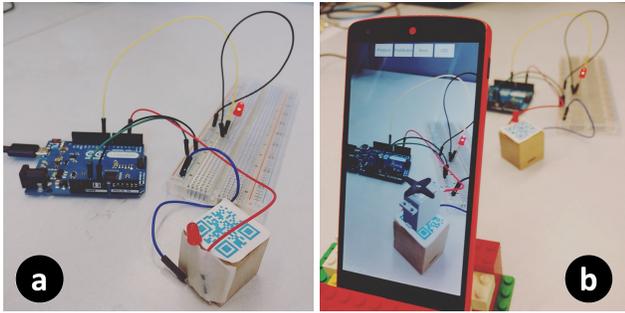


Figure 4. Polymorphic Cube: (a) trackable wooden cube connected to the Arduino and (b) companion AR application.

their physical computing projects despite missing electronic components. In addition, unlike these systems, the PMC system works with real circuits and simulates the missing I/O components in situ. The simulated I/O components not only respond to on-screen interaction triggers (as in d.tools [15]), but are physically and functionally part of the circuit.

Augmented Reality Tools

Several researchers have explored AR as a way to tightly couple the physical and digital worlds using rich and interactive 3D content [5, 8, 20]. People use readily available technologies (e.g., personal mobile devices) to view and explore the 3D content, thereby, improving portability [4, 10].

AR Circuits [1] is a commercial educational app, which uses AR to build circuits without any electronics. Learners’ assemble printed-paper components (e.g. battery, wire, bulb, switch) together to build virtual circuits. Learners can interact with the components in their circuits, but cannot program the circuit [1]. LightUp [2, 11] is a low-cost AR application that recognizes the circuit behavior and gives live and interactive graphic feedback using AR technology. ConductAR [21] is an AR tool, which can recognize and analyze hand-drawn, printed, and hybrid conductive ink patterns. The augmentation helps users to understand and enhance circuit operation. MixFab [30] is a mixed-reality environment that helps users design objects in an immersive AR environment for 3D fabrication. The immersive AR environment enables creating objects, interacting with the virtual objects and the introduction of physical objects into the design of the object.

Like the above examples, PMC design also uses AR to blend the physical circuit with the simulated I/O components. However, the goal of AR in PMC is *not* to teach electronics [1, 11], or to enhance the circuit building activity [21], or to assist in the fabrication process [30]. We use AR to simulate missing I/O components so that makers can continue building physical interactive circuits despite material constraints.

POLYMORPHIC CUBE: TECHNOLOGY PROBE

We implemented a technology probe [16], Polymorphic Cube (PMC) based on our vision for AR-mediated prototyping. The goal of PMC is to elicit feedback from makers about AR-mediated prototyping. To that end, PMC is simple and currently allows people to assign AR components, build and program circuits, and interact with the AR components (topics previously discussed in our vision).

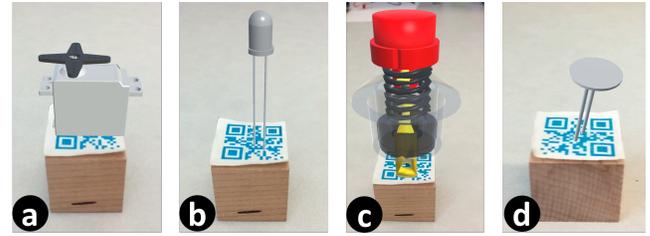


Figure 5. PMC prototype currently includes four electronic components: (a) servomotor, (b) LED, (c) pushbutton, and (d) photocell.

Design Overview

PMC consists of two main parts: a low-fidelity trackable physical cube (Figure 4a) and a companion smartphone AR application (Figure 4b). The cube, a physical placeholder for the missing electronic component, can be physically connected to a microcontroller pin (here, an Arduino), to facilitate building a physical circuit. The companion AR application allows the maker to select and assign PMC to the missing electronic component (e.g., LED, servo, photocell). Makers can program the connected AR components and real components using the Arduino IDE.

Construct and Assign AR Components

PMC is a very simple low-fidelity prototype – a wooden cube with a QR code and LED attached with tape (Figure 4a). The cube is a durable and a stable placeholder to connect to a circuit. We use a one-inch cube for consistent tag recognition. While our PMC is simple, the use of at hand materials such as paper-based AR tags, a found placeholder object, and a low-cost electronic component, to build an AR component means other people can build their own PMCs within material constraints.

A companion mobile application allows the maker to assign the cube to a variety of components (Figure 4b). Currently, PMC can simulate four components (Figure 5): LED (digital output), servo (analog/digital output), pushbutton (digital input), and photocell (analog input). The maker can select and assign the cube to different I/O components through a button-based menu at the top of the mobile interface. We use the concept of polymorphism to assign the cube to a variety of electronics. For visual interface design, polymorphism is said to be an essential property for keeping an interface simple [7]. Our current prototype achieves polymorphism by enabling the maker to author the stand-in component. Rather than having a large set of pre-assigned QR codes assigned to different I/O components, the maker can assign a QR code to any I/O component they need.

Build Circuit and Write Code

One face of the cube has an LED with two wires soldered to the LED legs. The wires connect the cube to an Arduino pin (Figure 4a). Connecting a component using wires is analogous to connecting traditional electronics. The simplistic two-wire connection reminds the maker that a component has to be connected to a specific microcontroller pin to program the component just as with using real components. The LED also serves as a feedback mechanism, indicating to the maker that the cube was connected correctly – if the wires are correctly

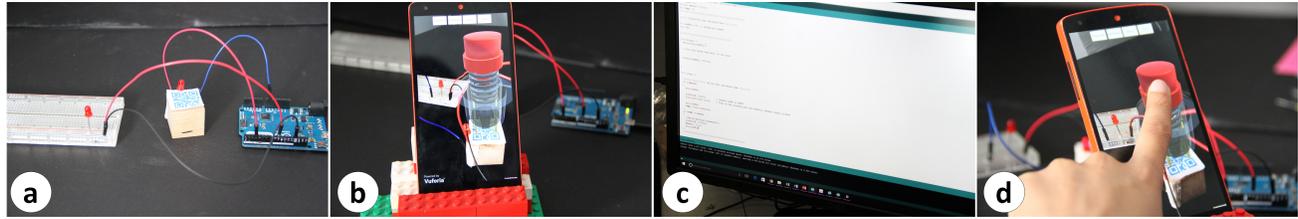


Figure 6. Example of a simple light controller built using PMC: (a) build a circuit, (b) assign the PMC cube to a pushbutton, (c) write code, and (d) interact with the AR pushbutton to test the system.

connected to the Arduino, the LED turns on. To program the AR components, the maker uses Arduino serial commands.

Test Circuit

PMC implements direct touch-based interaction with the virtual I/O components (Figure 3b). Direct touch with the virtual I/O components mimics interaction metaphors that a maker would use to interact with real components, thereby taking advantage of learned hands-on skills [17]. Each AR component model resembles the form of real component and affords similar interactions as the real components. For example, for a push button, the maker can push the cap of the 3D button model using touch on the screen. To visualize a button press, the button spring is animated to compress and expand.

Example

Figure 6 demonstrates an example of using PMC to build a simple prototype of a light controller system. In this example, PMC simulates a missing pushbutton. A maker completes four steps to prototype. First, the maker builds a light switch circuit (Figure 6a). For this, the maker connects the PMC cube to the breadboard and the microcontroller using the two wires attached to the cube. Second, the maker assigns the PMC cube to a pushbutton (Figure 6b) by selecting from the button-based menu positioned at the top of the screen. Third, the maker writes code to control the real-world LED using a virtual button and uploads the Arduino program to the Arduino Leonardo via USB (Figure 6c). Lastly, the maker interacts with the simulated pushbutton on the mobile screen by pressing and releasing the button cap to turn the light on/off (Figure 6d).

Implementation

Implementation of the PMC prototype system includes the following two main parts:

Hardware

PMC hardware has the same circuit footprint as an LED. Therefore, the microcontroller can detect the AR component. Our current implementation, does not use a smart breadboard to analyze the entire circuit. As a result, electrically, real-world components do not effect the AR components. However, because the microcontroller can detect the AR component, their behavior can be effected by the real-world components via programming. We configure a specific digital pin to be the PMC cube pin (as shown in the code snippet below).

```
int state = 1;
boolean detect = false;
int temp = 1;

void loop() {
  if (!detect)
  {
    delay(1000);

    // default mode is INPUT
    pinMode(3, INPUT);
    // Turn on the internal pull-up resistor
    digitalWrite(3, HIGH);

    delay(1000);
    temp = digitalRead(3);
  }
  if (temp != state)
  {
    // AR component connected to pin 3
    pinMode(3, OUTPUT);
    detect = true;
    delay(100);
  }
}
```

Software

All of the AR electronic component models used we downloaded free from the 3D warehouse website¹. We used the Vuforia unity SDK² to create the Android AR application. We wrote a C# program to establish communication between the Arduino and the companion smartphone application. The C# program communicates with the Arduino via a serial port. The communication with the Android application is via Wi-Fi.

The maker programs the behavior of the entire circuit, both AR and real-world components, using the Arduino IDE. To simulate an output AR component (e.g., servo), the C# program gathers the state value of the component (e.g., servo's position) from the Arduino code in real-time and writes them to a file on the webserver. The values written to the file are then used to simulate the AR component (e.g., rotate the servo). Similarly, for an input component (e.g., pushbutton), the C# program gets values via on-screen phone interaction ('0' for press down and '1' for release) and communicates that to the Arduino. The code snippet below shows the Arduino code used to turn on/off a physical LED via a simulated pushbutton.

```
if (Serial.available() > 0)
{
  // reading from the serial port.
  ch = Serial.read();
}
```

¹<https://3dwarehouse.sketchup.com/?hl=en>

²<https://developer.vuforia.com/downloads/sdk>

```

if (ch == '1')
{
  // button is pressed, LED on
  digitalWrite(ledPin, HIGH);
}
if (ch == '0')
{
  // button is not pressed, LED off
  digitalWrite(ledPin, LOW);
}
}

```

EVALUATING AR-MEDIATED PROTOTYPING

We conducted two studies, an observational lab study and an online questionnaire-based study, to evaluate PMC and to elicit feedback on the broader vision of AR-mediated prototyping. As part of the lab study, participants were introduced to PMC and they built simple prototypes of a controllable lamp using real-world prototyping materials and an AR component. We gathered feedback about usability and usefulness of PMC and the benefits and limitations of the broader vision of AR-mediated prototyping for addressing material constraints. Next, in the questionnaire-based study, we specifically focused on understanding how people envision interacting with AR components. In this section, we first detail the observational lab study method, followed by a description of the questionnaire-based study method. We present the results of both our studies in the next section.

Observational Lab Study

The primary goal of this study was to observe if participants are able to build physical computing projects using a single AR component.

Participants

Twelve people from the ages of 20 to 44 (3 females, 9 males) participated in our study. We recruited via notices posted to local makerspaces, emails sent to university wide graduate student mailing list, and word-of-mouth recruitment. We recruited participants who had prior experience with building circuits using the Arduino and Arduino programming. We selected participants on a first-come first-serve basis and the participants were remunerated with \$20. Our participants self-identified as makers and came from different disciplines and professional backgrounds: energy teacher, IT/ electronics consultant, visual artist and science communicator, and graduate students (electrical engineering, computational media design and computer science). Participants had a range of self-rated expertise in physical computing: novice (1), beginner (5), competent (5) and expert (1). Participants also had varied frequency of involvement with physical computing activities: rarely (4), occasionally (3) and frequently (5).

Experimental Setup

We conducted the study at a desk with a PC connected to dual displays and used one portion of the desk as the activity area. We taped an Arduino Leonardo to the activity area of the desk. In addition, we provided the participants with a breadboard, several connection wires, a polymorphic cube, a Google Nexus 5 smartphone, a LEGO-based phone stand, and circuit diagram printouts. We placed the real electronic components used for

the study – 1 LED, 1 servo, 1 photocell and 1 pushbutton on another desk beside the participants' desk. The circuit diagram printouts corresponded to each of the four components used in the study. We pre-installed the PMC application on the smartphone used in the study. For the purpose of this study, the participant could assign the PMC cube to a virtual LED, servo, photocell, and pushbutton. We provided sample code for all of the components. To help participants copy and paste code, we displayed the Arduino IDE on one monitor and the sample code on the second monitor. We placed a HD camera overlooking the activity area to video record all of the sessions. The researcher sat next to the participants' table to observe and to take notes of participants' interaction experience.

Procedure

We ran participants through the study individually, with each session running for about an hour. We introduced each participant to the goal of the study and encouraged them to talk-aloud throughout the study, expressing their thoughts about PMC. The study consisted of the following parts:

Pre-study questionnaire: Each participant completed a pre-study questionnaire that asked personal demographic information and a few questions regarding physical computing (i.e. prior experience, frequency of involvement, and things they had previously built). In addition, the questionnaire asked two questions about how participants addressed the challenge of missing electronic components during electronics prototyping.

Familiarization phase: After participants filled out the pre-study questionnaire, we conducted a step-by-step demonstration of building a pushbutton controlled servo circuit. For the demonstration, we used a physical pushbutton and the AR component was the servo. We first walked-through circuit building. After circuit building, we demonstrated programming. We copy-pasted the appropriate code samples into the Arduino IDE and uploaded the code to the Arduino board. Finally, we demonstrated the working system.

Task phase: After the familiarization phase, the participant was involved in the task phase. For this study, the goal of the task phase was to build a controllable lamp (a modified version of the Arduino *Touchy-Feely Lamp*³ starter kit activity) using an LED (output), servo (output), photocell (input) and a pushbutton (input). The task phase consisted of two parts:

(A) Sketch phase: In this phase, we asked the participants to sketch four unique designs of a controllable lamp. The participants used a task sheet to illustrate their ideas. Because the task was to build a controllable lamp, the use of an LED was consistent across all the designs. To create the scenario of "a missing component" we took each sketch and selected one component to be an AR component.

(B) Building phase: After the sketch phase, the participants built each design one at a time (total 4 trials). For each trial, the participant was always required to simulate one missing I/O component using PMC. There was no time limit for the trials. When the prototype was working as proposed in the

³<https://www.youtube.com/watch?v=AtplYQKyB5A>

design, the participants did a quick demo of the system for the researcher and then moved to the next trial.

Post-study questionnaire: At the end of the task phase, we asked participants to fill out a post-study questionnaire about their overall thoughts on using the PMC. We asked them to rate PMC using a 5-point Likert-scale [19] (where 1 was much worse and 5 was much better) on questions related to ease of use and experience with different components.

Post-study interview: The study concluded with a semi-structured interview intended to explore aspects of AR-mediated prototyping such as design iterations, multiple PMCs, and sharing.

Data Analysis

We transcribed individual interviews for posterior qualitative analysis [27]. From responses to each interview question, we made a list of higher level-themes (e.g., easy access and flexibility, interactivity supports sharing) as related to each of our design goals. We quantitatively analyzed the Likert-scale questionnaire to compute the median values. We used the individual video recordings to identify and count the different interaction styles used by the participants when interacting with the AR object.

Questionnaire Study

The primary goal of this study was to learn about how people prefer to interact with AR components (Goal #3). This study was conducted after the lab study.

Participants and Procedure

We asked our pilot participants and participants from our first lab study to take part in the online questionnaire. Thirteen participants responded.

We asked the participants to fill out an online questionnaire comparing the four different interaction styles for AR-mediated prototyping (shown in Figure 3). The questionnaire asked participants to rank the four interaction styles in order of preference (where 1 was least preferred and 4 was most preferred). We also asked participants to provide a rationale for their ranking order. The questionnaire consisted of eight categories of electronics: (1) light sensors, (2) weather sensors, (3) flex, force and vibration sensor, (4) direction sensors, (5) distance sensors, (6) sound sensors, (7) biometric sensors and (8) encoders. The categories are based on existing classification of electronics on online electronics store such as SparkFun and Phidgets. We analyzed the questionnaire quantitatively to compute the mean values. We qualitatively analyzed the participants comments to understand their rationale.

RESULTS

In this section, we discuss results from both our studies in the light of our goals and tasks of AR-mediated prototyping.

Responding to Lack of Materials

The meta-level goal for AR-mediated prototyping is to help makers' overcome the lack of material resources (e.g., electronic components). Although our motivation for lack of electronics comes from more extreme resource-constrained

contexts such as, an impoverished school in Brazil [25] and India [26], we found lack of materials to be a potential roadblock even when participants had easy access to material resources.

From our pre-study questionnaire, we found 11 of 12 participants living in North America had previously encountered a situation when they had limited access to electronics. Overall, participants had three practical solutions to overcome the challenge of material unavailability. First, the majority of our participants (8 of 12) had placed an order online and had decided to wait while the electronics arrived: "Usually I stop what I am doing and order the component. This can be difficult as it can delay the project for weeks sometimes" [P8]. Second, a few participants (4 of 12) had attempted to re-use existing and readily available electronics as an alternative: "I missed some switches in my design which I ended up replacing with transistors and resistors combined" [P7]. Lastly, one participant mentioned that they had borrowed the missing electronic components from a friend.

Using PMC, all participants agreed that they could continue to build projects despite missing components. Overall, all participants found that PMC is easy to understand (median=4, n=12) and was usable for the given task (median=4, n=12). Four participant quotes explicitly highlight the usefulness of PMC in responding to a lack of materials:

"if I open my box at home I have all of these [button and LED], but I don't have a servo. So it was cool that we started with the servo, because I actually tried what I could have done at home. Because I did not have a servo, I simulated it and it was really helpful" [P1].

"sometimes you want to try a part, but you have to order it. By the time it gets shipped, you may have forgotten the idea or had different ideas. It would be awesome if it could be as easy as going to DigiKey and download a model and play with it using this [PMC]" [P6].

"an alternative path to something like this would be a complete simulation but, this is more fun then watching on the computer. So I think this is a brilliant idea!" [P5].

"could save a lot of money by not having to buy all the different components you would need for design" [P8].

Building with AR Components (Goal #1, Task #1)

In our vision, we suggested that technology for AR-mediated prototyping should allow makers to build AR components (Goal #1) and this in turn can help makers with experimenting with a variety of components (Task #1). In this section, we discuss participant feedback related to this goal and task.

We envisioned that both physical and virtual forms of an AR component are important. From our lab study 4 of 12 participants suggested that physical form of an AR component is less important. Participant P4 argued that in AR-mediated prototyping, components do "not need to have a lot of physical presence". Participant P4's rationale was that because AR components are virtual we do not have to worry about the physical space they occupy – "one does not have to worry about things like if a motor has place to spin". Participant P4

suggested that the placeholder objects could be small discs or even a cable. A similar opinion was expressed by participant P10, who mentioned that when used in prototyping, the placeholder object could be shrunk, reduced to pieces of paper, or “*even integrated into the breadboard, so the breadboard would have specific pins that were cubes*”. Participant P12 added that if a placeholder object were used then “*it would be nice to have the cube comparable to the real component size*”.

Specific to virtual form, in our implementation, we used realistic 3D models of electronic components. Participants P6 and P9 specifically mentioned that the ability to have a 360 degree view of 3D models made the experience feel physical. One participant however, suggested that “*maybe it would be interesting to choose to go more abstract, because you can't really go fully realistic*” [P12].

In our current PMC design, we treat each AR component as a single electronic component (1-to-1 assignment). While seven participants agreed with this view, five participants suggested assigning one placeholder object to multiple components or sub circuit. Participant P3 suggested that when using multiple AR components the circuit could become “*bulky*”. Therefore, to “*cut the space*” they suggested that it might be better to assign a single placeholder object to multiple components. Participants P11 and P13 mentioned that 1-to-many assigning could be beneficial when the placeholder object represents multiple instances of the same components (e.g., strip of LEDs or buttons). One participant specifically mentioned that assigning one placeholder object to multiple components could be useful to better facilitate interactions with the components: “*Let's say, we had a big circuit here and we had 5 cubes spread. Then maybe at some point I would like to turn on a switch and cover a photocell, and press another button. So I don't know how to deal with it*” [P7].

Specific to experimentation with components, overall, all participants found PMC easy to assign and experiment with different components: “*I liked being able to change to anything very easily. That was the main thing I liked about it [PMC]*” [P12]. All our participants also mentioned that they could imagine using a tool like PMC to test their design ideas if the tool included access to a large library of electronic components. Participant P13 specifically mentioned that such a library need not be limited to existing electronic components and should include end-user defined components.

Coupling AR and Real-World Components (Goal #2)

We suggested that technology for AR-mediated prototyping must should allow makers build a unified circuit by coupling the AR and real-world components via circuitry and programming. In this section, we discuss feedback related to both circuitry and programming.

All participants successfully built four unique prototypes of the controllable lamp using PMC during the lab study. Overall, we found that participants with varying expertise level found it easy to build circuits using PMC (median=4, n=12). Participants could easily connect the cube to their circuits using the two wires attached to the cube.

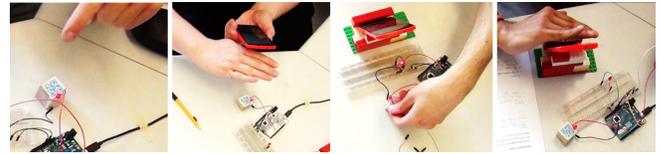


Figure 7. Observed examples of participants attempting to interact with the physical placeholder object and built in phone sensors.

From post-study interviews, we learned that 4 of 12 participants liked the simplistic two-wire connection. Participants P10 and P11 mentioned that the two-wire connection was easy and had an advantage over complex real components which require more knowledge and effort when (re)building a circuit. Participant P9 mentioned that simple wiring also helps aesthetics – “*its kind of cleaner*”. Although the goal of our vision is to support prototyping within material constraints, participant P1 considered PMC also to be a useful educational tool. They suggested that for educational applications it might be important to support wiring setup similar to real components: “*Let's say we had a transistor which is a three pin, then you may have mislead that it was only a two pin connection*” [P1]. In our current implementation, participants do not need to add resistors for the virtual components. Participant P11 explicitly mentioned that not worrying about details like resistors helps makes prototyping sometimes easier. As a suggestion for improvement, participant P1 mentioned, “*In general this [PMC] looks pretty good. Maybe in the software you can have some design rule check, for example connected to a node*”.

Related to programming, we found that participants with varying expertise level found it easy to program the AR components using the Arduino IDE (median = 4, n=12). Participant P12 explicitly mentioned that AR-mediated prototyping is helpful for: “*prototyping code; to see if it would do what you wanted, especially when you have complex code and want to see if your output would work correction*” [P12]. Unlike our vision that suggested leveraging makers learned skills with current physical computing platforms, one participant suggested using a tangible programming approach to AR-mediated prototyping: “*if there was a way to get rid of programming, then it [AR-mediated prototyping] would be even more physical*” [P9]. Participant P9 suggested using an approach similar to the technique of program by example demonstrated in projects such as Curlybot [14] and Topobo [23].

Interaction with AR components (Goal #3)

In our vision, we considered four interaction styles. In PMC, we implemented touch-based interactions with the AR objects (Figure 3b). To understand how people generally prefer to interact with AR components we looked to our study data from both the lab study and the questionnaire-data.

During the lamp-building task in the lab study (first study), we observed that 7 out of 12 participants attempted to interact with the simulated I/O components using built in phone sensors and direct interactions with the physical placeholder object (Figure 7). However, in contrast post-study responses from the lab study highlighted that participants liked the touch-based interactions.

Responses to the the Likert-questionnaire from the lab study revealed that participants found interacting with a virtual servo, photocell, and pushbutton using touch about the same as interacting with their real-world counterparts (median=3, n=12). Specifically for input components, post-study interviews revealed that all participants liked the simple touch-based interactions. Participant P9 explicitly mentioned that the animation of physical components provided useful feedback for interaction: *“With the physical one [button], it was like did I get it, did I press it on. With this [virtual button] I knew it was working, the feedback was really nice”* [P9].

This pattern was also observed in the responses to online-questionnaire study, which included a wider variety of electronic components (sensors and encoders). The responses to the online questionnaire showed that in order of preference, the majority of the participants first preferred either touch-based interaction with the virtual component or interactions using widgets (median=2.5, n=13), a close second choice was built in phone sensors (median=2.25, n=13), and direct physical interaction with the placeholder object was least preferred (median=1.94, n=13).

Participants reasoned that they preferred touch-based interaction with the virtual components because it is intuitive, the interaction was collocated with the object, it facilitated more control, and that it could be consistently used with a variety of electronics. Interacting with widgets was preferred because participants had prior experience using widgets for controlling specific values.

Although interaction using built in phone sensors was a close second choice, all participants commented that because phones had limited sensors, this interaction style would be less consistent. One participant added: *“this seem like a tricky option because it can create a disconnect between how users interact with components. If I’m acting within a virtual world for one component (say a button), it doesn’t seem consistent that the phone acts as a sensor for another (why do I have to shake the phone for a vibration sensor/accelerometer?)”* [P13].

Similarly, a reduced degree of coherence (i.e. the degree to which physical and digital might be perceived as the same thing) [18] was highlighted as a possible problem for interacting directly with the placeholder object: *“I feel like its more appropriate for a full virtual reality environment. In the current AR setting, it seems like it would divide your attention between two objects (the AR device, and the actual circuit). I prefer focusing on one thing at a time, so it makes sense to me to keep all interactions virtual”* [P13].

Social Interactions (Goal #4, Task #3)

We did not implement a function to share code and components in our current version of the PMC. However, our participants reflected on this aspect during the interview. We gathered that the participants envisioned two types of sharing: sharing software code, and sharing software code and electronics.

In the *Sharing Software Code* model of sharing, participants (5 of 12) suggested that every individual maker could own a personal placeholder object, but the software code mapped to the placeholder object could be shared as a community

resource. Some of the suggested benefits of this approach included sharing code in classroom settings (P6), and the ability to download models of electronics and use code written by others to support implementation tinkering (P8, P12). In addition, this model was said to work well for remote collaborative practices (P12).

In the *Sharing Software Code and Electronics* model of sharing, participants (5 of 12) suggested that the maker could share the placeholder object as a single unit comprising code and component. Some of the suggested benefits of this approach included: allowing makers to get quick help from experts (P4, P5), encouraging open-source sharing of code (P7, P13), and enabling exchange of complex physical systems such as a tangible puzzle game, with others (P13).

Transition to Real-World Components (Task #2)

In our lab study, design refinement took the form of rebuilding four unique versions of a controllable lamp. We did not explicitly focus on transitioning to real-world components. However, in each trial participants had to swap out AR components with real components. Two participants (P10, P12) explicitly mentioned that transitioning between AR components was easy: *“It [PMC] was used in different situations. It [PMC] did not have to be reconfigured. Only programming had to be changed. So physically it is definitely an advantage over the physical components”* [P10].

Participant P5 added, that PMC-like tools are useful during the *“experimentation and playful phase”*. However, they suggested that after the prototyping phase is complete and when making actual designs, the maker has to use a variety of tools for making: *“this [PMC] is another useful tool and you would use it along with all the other tools”* [P5].

DISCUSSION AND FUTURE WORK

The primary goal of our AR-mediated prototyping approach is to enable makers to use technology to continue building physical computing projects despite missing material resources. All our participants’ could overcome the lack of a required I/O component and build several prototypes of the controllable lamp using PMC. Initial reactions of makers toward PMC have been encouraging. Participant statements not only reveal that our vision considerations were meaningful, but they also demonstrate a high level of excitement toward the use of PMC-like technology for making when challenged with material shortages. However, PMC, as a first exploration in this direction, also raises some questions to be explored in the future.

Physical Kinematics. Participants found PMC useful for overcoming material lack, prototyping physical computing ideas, fearlessly exploring electronics, and sharing code and electronics. However, one limitation of our current implementation of PMC is that AR components cannot physically demonstrate material behaviors. In the future, it would be interesting to explore the use of low-cost self-actuated flexible interfaces (e.g., [24]) to enable physical kinematics.

Scalability. In our current study of PMC, we allowed participants to use one cube as a stand-in for one missing electronic

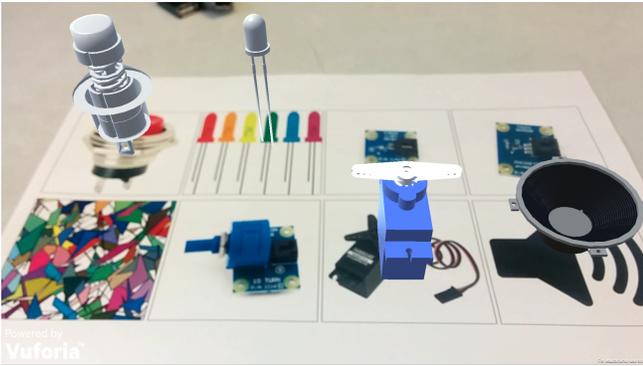


Figure 8. Test for detecting multiple AR components (button, LED, servo, and speaker) simultaneously.

component. Our findings indicate that participants could successfully work with one augmented cube when building physical circuits. However, in scenarios where a maker may not have immediate or easy access to many materials, scalability of the technology is important. From an implementation standpoint, we have successfully tested tracking multiple objects (Figure 8). However, as indicated by our participants, there are several aspects to consider to scale interaction when using multiple AR objects. For example, if two placeholder objects are placed far away from each other and the maker needs to interact with them simultaneously, then the maker would need a much larger display than a phone can offer. One solution is to make use of larger displays. It would also be interesting to explore solutions similar to Surround-See [32] that enable peripheral vision around mobile devices. In addition to exploring how to scale AR-mediated prototyping, an important thread to explore in the future is understanding how many AR components can be used in circuit before AR-mediated prototyping begins to deviate away from physical making.

Open-source 3D models. Our participants appreciated the flexibility of AR-mediated prototyping. All participants mentioned that having an elaborate list of virtual components would improve the usefulness of PMC-like tools – “...I see myself using it [PMC] if you had a library of models” [P5]. In our current PMC prototype, we used freely available online 3D models of electronics. Digital easy-to-use maker tools for creating models of electronic components would help expand the ecology of virtual materials that makers can use within their hybrid AR-mediated physical computing projects. For example, participant P13 suggested that support for design iteration could be improved further by allowing participants to use both physical and virtual components (e.g., virtual knobs, screens, and UI components). In addition, makers could share their digital design files along with software code on online communities such as Thingiverse⁴ or Instructables [31] to help others explore ideas by tinkering code and material functionality.

Transferability of Skills. Interviews with participants raised questions about skill transfer. One participant mentioned that the simplistic two-wire connection can be misleading and it might be initially difficult to understand the abstraction of the

cube. To enable transferability of skills, PMC could use AR to provide information about missing physical materials. Similar to works like LightUp [11] and ConductAR [21], when makers replace the surrogate AR component with a real component, the AR application can help makers with building circuits, optimize circuits and also help find errors and correct errors. This could help makers overcome both material challenges as well as conceptual difficulties involved in technology-based DIY.

Interactions with Virtual Components. Likert-questionnaire responses revealed that there was no clear majority for the preferred interaction style. For example, while majority (10 of 12) of the participants said that the touch-based interaction (Figure 3b) with a pushbutton felt the same as real-world interaction with a pushbutton, 5 of 12 participants found touch-based interaction less satisfying for the photocell. We used touch-based interaction technique in the current PMC to provide a consistent interaction metaphor. In the future, it would be interesting to explore which AR technique is best for different categories of components (sensors, actuators, and encoders).

CONCLUSION

In this paper, we presented our vision for technology for AR-mediated prototyping for physical computing projects. The goal of AR-mediated prototyping is to help makers continue to build physical computing projects despite missing material resources. To demonstrate and evaluate our vision we implemented a technology probe, Polymorphic Cube (PMC). The goal of PMC was to provoke thoughts about making despite missing electronics, and to elicit feedback about PMC in the light of our vision. We conducted two studies to gather participants’ reflections about AR-mediated prototyping. We found PMC could help makers focus on prototyping project ideas instead of researching for alternative materials. In addition, makers can continue to take part in implementation tinkering and testing of multiple design ideas. One limitation of our hybrid AR-mediated prototyping is that the AR components are virtual and cannot physically affect real components. However, participants’ reaction highlights that this might be a negligible limitation given that makers can continue to prototype rather than discarding ideas or forgetting ideas while waiting for components to become available. We encourage future researchers to examine how our findings and design goals apply to other instances of technology for AR-mediated prototyping.

REFERENCES

1. AR circuits. 2016. AR Circuits - Augmented Reality Electronics Kit. <http://arcircuits.com/>. (2016). [Online; accessed 2016].
2. Zain Asgar, Joshua Chan, Chang Liu, and Paulo Blikstein. 2011. LightUp: a low-cost, multi-age toolkit for learning and prototyping electronics. In *Proceedings of the 10th International Conference on Interaction Design and Children*. ACM, 225–226.
3. Autodesk. 2016. Autodesk Circuits - Electronics from beginner to pro. <https://circuits.io/>. (2016). [Online; accessed 2016].

⁴<http://www.thingiverse.com/>

4. Ronald Azuma, Yohan Baillot, Reinhold Behringer, Steven Feiner, Simon Julier, and Blair MacIntyre. 2001. Recent advances in augmented reality. *IEEE computer graphics and applications* 21, 6 (2001), 34–47.
5. Ronald T Azuma. 1997. A survey of augmented reality. *Presence: Teleoperators and virtual environments* 6, 4 (1997), 355–385.
6. Jonathan Bean and Daniela Rosner. 2014. Making: Moevement or Brand? *interactions* 21, 1: 26–27. (2014).
7. Michel Beaudouin-Lafon and Wendy E Mackay. 2000. Reification, polymorphism and reuse: three principles for designing visual interfaces. In *Proceedings of the working conference on Advanced visual interfaces*. ACM, 102–109.
8. Mark Billinghurst, Raphael Grasset, and Julian Looser. 2005. Designing augmented reality interfaces. *ACM Siggraph Computer Graphics* 39, 1 (2005), 17–22.
9. John J Burke. 2014. *Makerspaces: A practical guide for librarians*. Vol. 8. Rowman & Littlefield.
10. Julie Carmigniani, Borko Furht, Marco Anisetti, Paolo Ceravolo, Ernesto Damiani, and Misa Ivkovic. 2011. Augmented reality technologies, systems and applications. *Multimedia Tools and Applications* 51, 1 (2011), 341–377.
11. Joshua Chan, Tarun Pondicherry, and Paulo Blikstein. 2013. LightUp: an augmented, learning platform for electronics. In *Proceedings of the 12th International Conference on Interaction Design and Children*. ACM, 491–494.
12. Bettina Conradi, Verena Lerch, Martin Hommer, Robert Kowalski, Ioanna Vletsou, and Heinrich Hussmann. 2011. Flow of electrons: an augmented workspace for learning physical computing experientially. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*. ACM, 182–191.
13. Emmanuel Dubois¹², Laurence Nigay, Jocelyne Troccaz, Olivier Chavanon, and Lionel Carrat. 1999. Classification Space for Augmented Surgery. an Augmented Reality Case Study. In *Human-Computer interaction, INTERACT'99: IFIP TC. 13 International Conference on Human-Computer Interaction, 30th August-3rd September 1999, Edinburgh, UK*, Vol. 1. 353.
14. Phil Frei, Victor Su, Bakhtiar Mikhak, and Hiroshi Ishii. 2000. Curlybot: designing a new class of computational toys. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 129–136.
15. Björn Hartmann, Scott R Klemmer, Michael Bernstein, Leith Abdulla, Brandon Burr, Avi Robinson-Mosher, and Jennifer Gee. 2006. Reflective physical prototyping through integrated design, test, and analysis. In *Proceedings of the 19th annual ACM symposium on User interface software and technology*. ACM, 299–308.
16. Hilary Hutchinson, Wendy Mackay, Bo Westerlund, Benjamin B Bederson, Allison Druin, Catherine Plaisant, Michel Beaudouin-Lafon, Stéphane Conversy, Helen Evans, Heiko Hansen, and others. 2003. Technology probes: inspiring design for and with families. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 17–24.
17. Hiroshi Ishii and Brygg Ullmer. 1997. Tangible bits: towards seamless interfaces between people, bits and atoms. In *Proceedings of the ACM SIGCHI Conference on Human factors in computing systems*. ACM, 234–241.
18. Boriana Koleva, Steve Benford, Kher Hui Ng, and Tom Rodden. 2003. A framework for tangible user interfaces. In *Physical Interaction (PI03) Workshop on Real World User Interfaces*. 46–50.
19. Rensis Likert. 1932. A technique for the measurement of attitudes. *Archives of psychology* (1932).
20. Paul Milgram, Haruo Takemura, Akira Utsumi, and Fumio Kishino. 1995. Augmented reality: A class of displays on the reality-virtuality continuum. In *Photonics for industrial applications*. International Society for Optics and Photonics, 282–292.
21. Koya Narumi, Steve Hodges, and Yoshihiro Kawahara. 2015. ConductAR: an augmented reality based tool for iterative design of conductive ink circuits. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 791–800.
22. Dan O'Sullivan and Tom Igoe. 2004. *Physical computing: sensing and controlling the physical world with computers*. Course Technology Press.
23. Hayes Solos Raffle, Amanda J Parkes, and Hiroshi Ishii. 2004. Topobo: a constructive assembly system with kinetic memory. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 647–654.
24. Anne Roudaut, Abhijit Karnik, Markus Löchtefeld, and Sriram Subramanian. 2013. Morphees: toward high shape resolution in self-actuated flexible mobile devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 593–602.
25. Arnan Sipitakiat, Paulo Blikstein, and David P Cavallo. 2004. GoGo board: augmenting programmable bricks for economically challenged audiences. In *Proceedings of the 6th international conference on Learning sciences*. International Society of the Learning Sciences, 481–488.
26. Sowmya Somanath, Lora Oehlberg, Janette Hughes, Ehud Sharlin, and Mario Costa Sousa. 2017. 'Maker' within Constraints: Exploratory Study of Young Learners using Arduino at a High School in India. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. ACM, 96–108.
27. Anselm Strauss and Juliet M Corbin. 1990. *Basics of qualitative research: Grounded theory procedures and techniques*. Sage Publications, Inc.

28. Joshua G Tanenbaum, Amanda M Williams, Audrey Desjardins, and Karen Tanenbaum. 2013. Democratizing technology: pleasure, utility and expressiveness in DIY and maker practice. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2603–2612.
29. Uhling, Frank. 2016. react3D Electric-Tangible User Interface. <https://www.youtube.com/watch?v=6qTTQKWfV8Q>. (2016). [Online; accessed 2016].
30. Christian Weichel, Manfred Lau, David Kim, Nicolas Villar, and Hans W Gellersen. 2014. MixFab: a mixed-reality environment for personal fabrication. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems*. ACM, 3855–3864.
31. Eric Wilhelm and Saul Griffith. 2015. Instructables - DIY How To Make Instructions. <http://www.instructables.com/>. (2015). [Online; accessed 2015].
32. Xing-Dong Yang, Khalad Hasan, Neil Bruce, and Pourang Irani. 2013. Surround-see: enabling peripheral vision on smartphones during active use. In *Proceedings of the 26th annual ACM symposium on User interface software and technology*. ACM, 291–300.