PrototypAR: Prototyping and Simulating Complex Systems with Paper Craft and Augmented Reality

Seokbin Kang[1], Leyla Norooz[2], Elizabeth Bonsignore[2], Virginia Byrne[3], Tamara Clegg[2,3], Jon E. Froehlich[4]
Computer Science[1], College of Information Studies[2], College of Education[3], University of Maryland
Allen School of Computer Science and Engineering[4], University of Washington
{sbkang, leylan, ebonsign, vbyrne, tclegg}@umd.edu | jonf@cs.uw.edu

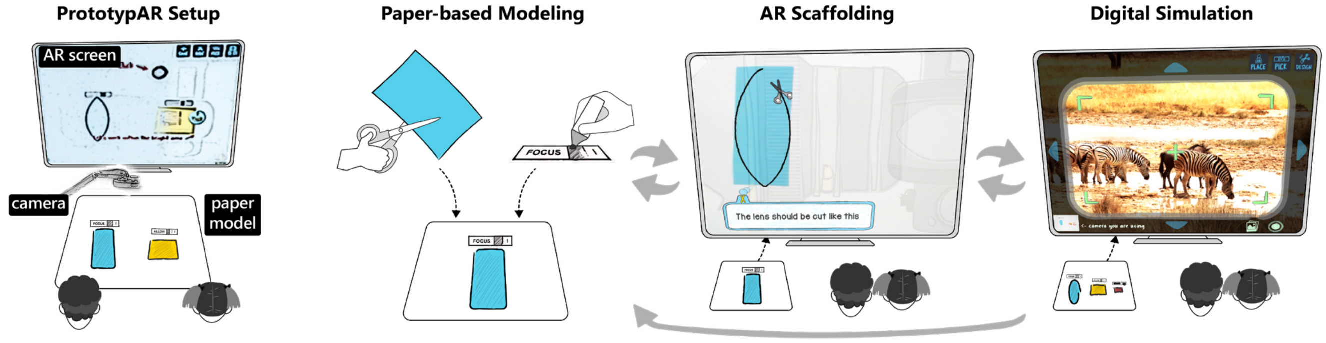

Figure 1. With PrototypAR, children can work together to create a complex system using paper craft, receive instant feedback about their design overlaid with augmented reality, and then test their design in a digital simulation environment. Above, two children create a camera lens by cutting out blue paper and sketching a bar graph to specify focal length, receive feedback about the shape, and then test their design by taking pictures in the simulation environment.

ABSTRACT

We introduce *PrototypAR*, an Augmented Reality (AR) system that allows children to rapidly build complex systems using paper crafts and to test their designs in a digital environment. PrototypAR combines lo-fidelity prototyping to facilitate iterative design, real-time AR feedback to scaffold learning, and a virtual simulation environment to support personalized experiments. Informed by three participatory design sessions, we developed three PrototypAR applications: *build-a-bike, build-a-camera,* and *build-an-aquarium*—each highlights different aspects of our system. To evaluate PrototypAR, we conducted four single-session qualitative evaluations with 21 children working in teams. Our findings show how children build and explore complex systems models, how they use AR scaffolds, and the challenges they face when conducting experiments with their own prototypes.

CCS CONCEPTS

Human-centered computing → Mixed / augmented reality

KEYWORDS

Tangible interaction; Augmented reality; Learning; Children

INTRODUCTION

Complex systems such as combustion engines and the human body are made up of interrelated components that interact to form a holistic, interdependent system [2,23]. Despite their pervasiveness in everyday life, complex systems are challenging to learn and to teach [13,38]. Prior work has shown that students struggle to understand how individual parts of a system affect the system's operation as a whole [49,57,69], narrowly focus on visible aspects like a system's structure [33], and have limited access to real examples that can affirm or contradict their understanding [3,13,38].

To address these challenges, prior work has explored the use of interactive computer-based simulations where children can build or manipulate aspects of a system and study differences in simulated results [15,18,38,55]. This approach allows learners to interact with otherwise inaccessible phenomena [33,38], helps reveal and correct their misconceptions [38], and improves their grasp of how a system functions as a whole [66]. However, existing approaches use traditional mouse-and-keyboard interfaces that limit how models are constructed, do not scaffold learners through the full design process—from modeling to experimentation, and are typically designed for older children (*e.g.,* middle school and beyond).

In this paper, we introduce *PrototypAR*, an AR-based "*smart desk*" that allows children to prototype complex systems using familiar paper crafts, to learn about and correct mistakes via AR-based feedback, and to test their creations in a simulation environment (Figure 1). The tangible approach is intended to facilitate rapidly prototyping ideas [45] and to promote collaborative and playful experiences [59]. As a child builds a paper prototype, PrototypAR analyzes their work using computer vision and provides in-situ scaffolds via AR. These scaffolds provide design feedback [21,67] and bridge connections to existing knowledge to help children solve problems that otherwise

IDC '19, June 12–15, 2019, Boise, ID, USA

ACM ISBN 978-1-4503-6690-8/19/06…$15.00
https://doi.org/10.1145/3311927.3323135

might be too difficult [29]. At any point in the design process, children can choose to test their model in a virtual simulation environment. Because the testing environment is digital, there is broad flexibility in how a design can be simulated and used for scientific inquiry (*e.g.,* testing hypotheses).

As initial work, our research questions are exploratory: What is the interplay between physical prototyping, AR feedback, and virtual simulations? What are the key benefits and challenges of a “smart desk” approach for learning? What aspects of PrototypAR seem to support design practices and complex systems learning? To address these questions, we designed and developed PrototypAR through three participatory design sessions with 10 children. These sessions enhanced our understanding of how children approach design and experimentation in a mixed-reality environment. We also gained design ideas for AR-mediated scaffolds, including increased support for iterative design and experimentation. Across the sessions, we developed three PrototypAR applications for exploring scientific phenomena and engineering concepts: *build-a-bike*, *build-a-camera*, and *build-an-aquarium*.

To evaluate PrototypAR, we conducted four single-session studies with 21 children who designed and tested the *build-a-bike* and *build-a-camera* applications. Through a qualitative analysis of video recordings, questionnaires, and focus group interviews, we found that PrototypAR allowed children to progressively build complex system models and explore a breadth of designs. Using the AR design feedback and simulations, children were able to repeatedly evaluate their prototypes and examine how different designs influence a system’s function. However, children struggled with designing experiments and interpreting results, which led to partial understandings.

In summary, our contributions include: (i) a novel AR-based prototyping system for children that supports paper-based modeling and simulation of complex systems; (ii) findings from participatory design studies and user studies that illustrate how children can engage in iterative modeling and personalized experiments as well as identify opportunities and challenges; and (iii) reflections on a tangible modeling approach for children’s complex systems learning.

RELATED WORK

PrototypAR is informed by learning and design theories, educational technology for complex systems learning, and HCI approaches to interactive prototyping tools.

Theoretical Underpinnings

Children often see complex systems at a macro, "black box" level of visible inputs and outputs, so that knowledge of internal components and mechanisms remains inscrutable (*e.g.,* [33]). Our work focuses on helping children learn about complex systems through design and experimentation. We draw upon three theoretical constructs.

First, to raise children’s awareness of the interrelated elements that comprise complex systems, PrototypAR operationalizes the *Structure-Behavior-Function* (SBF) framework [23,33], which breaks complex systems into three parts: *structure*, elementary components and their relationships; *behavior*, how structure elements work individually and together; and *function*, the purpose of the system as a whole or its components. With this three-pronged approach, the SBF framework can systematically attune children's attention to fundamental aspects of complex structures and help them unpack and visualize the typically obscure connections between design and mechanics.

Second, PrototypAR enacts the SBF framework within a constructionist learning environment. *Constructionism* is a learning theory that emphasizes how learning happens "felicitously" when learners make and tinker with physical artifacts [31]. With PrototypAR, learners apply the SBF framework by designing and constructing component elements of larger systems, experimenting with their designs, exploring solutions, and receiving and sharing feedback.

Finally, to aid design tasks and children’s understandings of complex systems, we use software-mediated *scaffolding* [53], which offers pedagogical assistance via software tools. In general, scaffolding can take many forms from prompts and coaching to suggested task breakdowns—each which help learners accomplish work that may otherwise be too advanced. Ideally, as a child develops their skillset, scaffolds should be designed to seamlessly fade away [12,25]. We provide computer-mediated scaffolds that facilitate craft modeling, iterative design, and conceptual understanding.

Educational Technology for Complex Systems Learning

Prior educational technology aimed at complex systems learning can be broken down into three approaches: (i) interactive simulations such as *SimSketch* [6] and *NetLogo* [64] that allow learners to test their own ideas about complex systems; (ii) participatory simulations like *Hubnet* [68] and *Beesim* [50] in which learners enact the roles of elements in complex systems; and (iii) conceptual representations such as *SBFAuthor* [22] and *SBF Hypermedia* [42] that facilitate organizing and representing knowledge about complex systems. PrototypAR draws upon each of these approaches but differs in the use of paper crafts for modeling, the integration of computer vision and AR to provide real-time scaffolding, and the focus on elementary-aged learners.

Interactive simulation systems show promise in improving students’ conceptual understandings through experimentation [55,75]. To enable representing and testing ideas, existing systems offer modeling interfaces that generally follow one of three paradigms: (1) a direct manipulation interface where users drag-and-drop pre-defined primitives of a simulation [14,15,72,74]; (2) a sketch-based interface where users can draw entities to construct a system [6,70,71]; or (3) a programming interface where users specify behaviors of various types of entities [5,55,56]. While each paradigm has its advantages—for example, sketch-based interfaces can promote self-expression in modeling [6]—they also introduce challenges for novices in that each necessitates learning of application-specific modeling interfaces, limits opportunities for collaboration, or requires learners to have programming skills. Our work takes a tangible approach that uses craft materials—already familiar to children—to build models. We envision the tangible interface will facilitate representation of children's ideas and understandings [45] and promote collaborative learning.

Our approach for supporting tangible interfaces is not new. Physical manipulatives combined with digital feedback such as *Flow Blocks* [77] or *TimeBlocks* [29] have been considered particularly effective for children's learning. For example, research on the *Flow Blocks* system suggests its potential to scaffold children's ability in understanding an abstract concept of causal effects. *TimeBlocks* demonstrated that illuminated interactive blocks can facilitate children's communications about an abstract concept of time. PrototypAR is distinct in that it supports free-form modeling—rather than manipulating pre-existing tangible artifacts—and provides situated scaffolds via AR—to bridge knowledge gaps and help manage modeling tasks.

Interactive Prototyping Tools

HCI research has long focused on prototyping tools to support creative design [1], personal fabrication [61], and user interface design [40]. A key design tenet of these systems is to support tight, lightweight loops between creating and testing [28]. For example, *BOXES* [36] emphasizes the rapid creation of functioning prototypes using lo-fi materials such as cardboard and aluminum foil and immediate testing to support iterative design. A second key tenet is providing contextual guidance and support [43]. For example, Marner *et al.* [44] suggests projecting visual guidance onto the surfaces of an on-going foam prototype to help produce a specific model. In PrototypAR, we explore how the iterative creation and testing of paper models could contribute to learning and examine how children react and use AR-mediated guidance.

DESIGN PROCESS

To design PrototypAR, we used an iterative, human-centered design process that included participatory design activities with children and adult designers. We first highlight three overarching design goals for PrototypAR, which were informed by prior work [37,54,73] and our own experience designing and evaluating children's learning tools.

- **Support engineering design.** We aim to support the engineering design concept and practice of generating, testing, and refining designs, which is foundational in STEM education [17,47].
- **Embed computer-mediated scaffolding.** Scaffolds should assess children's current understandings and adapt to their needs [46].
- **Facilitate inquiry.** We aim to automate the steps of inquiry [11,16] (*e.g.,* designing experiments and collecting results, making interpretations).

Participatory Design (PD) with Children

We co-designed PrototypAR using a participatory design method called *Cooperative Inquiry (CI)* [24] that is useful to understand how a technology fits children's needs and abilities, collect feedback about the technology, and generate design ideas. In partnership with an on-going design group, we conducted three CI sessions with 10 children (ages 8-11) and six adult design partners. In each session, groups of children and adults worked together as equal partners to brainstorm and elaborate upon each other's ideas from conception to production. To help participants understand the concept of PrototypAR, we used the technology immersion technique [35]. We had participants use an early prototype and examined: (i) How do children approach paper-based modeling in an AR environment? (ii) What do children find difficult to use or understand with PrototypAR? (iii) What types of scaffolds do children need for modeling and experimentation?

Session 1: Children's Interaction with PrototypAR

To gain an understanding of how children interact with PrototypAR, we invited children to use an initial prototype of the *build-a-bike* application and share their ideas. After a brief introduction to PrototypAR (15-minutes), children and adult co-design partners spent 40 minutes using the system and offering their feedback in the form of "likes, dislikes, and design ideas". A researcher synthesized high-level findings and discussed them with the children and the adults.

Overall, we found that children were able to use PrototypAR to prototype models and conduct experiments. Based on observations and comments, children seemed to like the use of paper craft for modeling (e.g., "*making our own shapes*"), the responsive simulations (e.g., "*the gears mirror the paper size*"), and the personalized experiments (e.g., "*we can race our gears*"). After making prototypes, children tested them in the virtual simulation environment and observed how different designs affect the bike's performance. One group simulated three different prototypes and reported, "*The yellow [rear gear] is so small and it still won.*" Though children appreciated the usefulness of AR design feedback (*e.g.,* a child stated "*Yes it was helpful …[to] tell you where to move it*"), some complained that the scaffolds constrained their creative design (*e.g.,* "*It was picky*").

Session 2: Children's Design Ideas

In the second session, we asked children for ideas to improve the PrototypAR interface by building lo-fi prototypes. We used a *Bags-of-Stuff* [19] technique in which children use craft supplies (*e.g.,* fabrics, cardboard, markers) to communicate design ideas. Children presented their lo-fi prototypes and an adult partner synthesized the high-level themes therein. The following themes emerged (Figure 2): (i) highlight design errors early and at multiple stages of the design process; (ii) give users more control over design feedback (*e.g.,* when and at what level of detail); (iii) enable user control of "invisible" properties of a complex system (*e.g.*, exposure time for a camera shutter); (iv) enrich the prototyping experience with multimedia and multiple modalities (*e.g.,* speech interface, sound, 3D VR goggles).

Figure 2. Example lo-fi prototypes from the second PD session, including: (a) integrating testing views at multiple stages of design; (b) allowing for user control to receive the design feedback; and (c) providing control of invisible attributes (*e.g.,* exposure time).

Session 3: Challenges and Scaffolds for Learning

Finally, to identify what aspects children found difficult with complex modeling tasks and to elicit ideas for scaffolding, we conducted a session using the more complex *build-a-camera* application. Before the session, we incorporated design ideas from previous sessions into PrototypAR, including: adding a hint button to allow children to control how and when they receive feedback as well as additions to the prototyping interface to enable design of component behaviors (*e.g.*, focal length of a lens). In this session, only one of the three groups succeeded in creating a complete prototype; the others were overwhelmed by the large number of design options involved in modeling the camera system. Because of their struggles, both children and adults suggested ideas to better scaffold learners, including: (i) focusing users' work on one design element at a time; (ii) prompting users to switch between making and testing; (iii) suggesting different options to encourage divergent design; and (iv) assisting users in setting up comparisons between prototypes in the simulation environment.

PROTOTYPAR SYSTEM

PrototypAR operates in two modes: *AR design* mode and *experiment* mode. In the *AR design* mode, the user can prototype a complex system using lo-fi materials. PrototypAR actively tracks the work surface and offers adaptive *scaffolding* to suggest needed actions or provide corrective advice. At any time, the user can switch to *experiment* mode to make observations about *how* their prototypes function and *why* through virtual simulations.

PrototypAR Design

PrototypAR is comprised of: (i) a *lo-fi prototyping interface* to support rapid creation of complex systems models; (ii) *AR scaffolds* to assist design tasks and learning; and (iii) *virtual simulations* to enable experimentation with prototypes.

Lo-fi Prototyping Interface

The prototyping interface allows children to model complex systems using paper crafts. To promote understanding through design, PrototypAR supports SBF modeling where the user models the structural elements and their behaviors that contribute to a complex system's overall function.

Designing structure. In PrototypAR, the representation of *structural elements* includes an object's type, shape, size, position, and relationship to other elements. The user designs a structural element by selecting a colored paper, cutting it into a shape, and arranging it on the augmented canvas. When beginning a design, PrototypAR augments the work surface with a structural outline of the target system (Figure 3). For example, in the *build-a-bike* application, a bicycle sketch is shown with key structural elements missing like the gears, pedals, and chain. The outline—which is visible on the AR display—serves as a visuo-spatial cue to aid the child in thinking about the shape and size of each component (*e.g.,* the gear should fit within the wheel) and location (*e.g.,* the gear should be at the wheel's center). To help the child think about and distinguish different structural elements, we map the paper's color to a particular object type (*e.g.,* the back gear is yellow while the front gear is green).

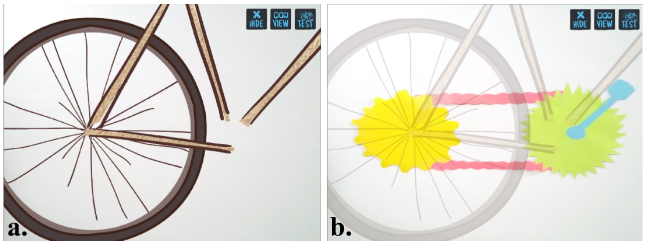

Figure 3. (a) The work surface is augmented with a design skeleton to help structural design; and (b) a final bicycle design with gears, pedals, and a chain.

Designing behavior. Because behaviors are more abstract and dynamic than structures, they are often more difficult to understand [32] and likely to be omitted in students' models [34]. In PrototypAR, children design behaviors explicitly via printed behavioral labels that are placed next to their corresponding structures. Each label has a *behavior name* and a *data field*, which can be filled in with marker to specify a behavioral variable (Figure 4). There are two label types: *numerical* and *categorical.* Numerical fields are specified by filling in a horizontal progress bar while categories are selected by filling out a check box. To help the user learn about and specify behaviors, the AR system augments labels with definitions and design instructions.

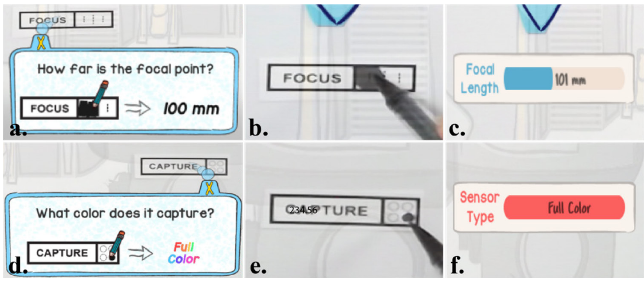

Figure 4. The behavioral labels are augmented with instructions to describe (a) a numerical value (*e.g.*, "how far is the focal point?") or (d) a categorical value (*e.g.*, "what color does it capture?"). (b, e) After the user fills in the label, (c, f) the system augments the label with a value.

AR Scaffolds for Prototyping

PrototypAR provides three types of AR scaffolds, which were informed by prior research [8,53] and our participatory design sessions: (i) supportive scaffolds to provide domain knowledge related to system models; (ii) procedural scaffolds to guide learners through the PrototypAR interface; and (iii) strategic scaffolds to facilitate the design process.

Supportive scaffolds. To help resolve misunderstandings and aid progress towards design completion [37], supportive scaffolds give children immediate feedback and hints on potential design problems. The scaffolds are dynamically generated based on real-time recognition of the user's paper prototype and pop-ups next to the target of interest using animation, images, and basic text. In total, PrototypAR provides six supportive scaffolds, including feedback for shape, position, and existence of an object. Three examples are shown in Figure 5.

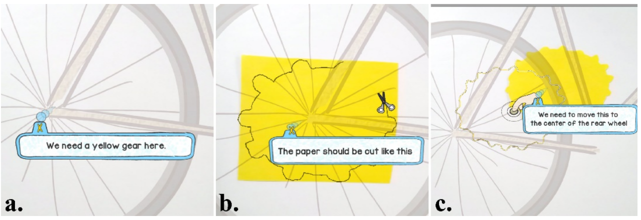

Figure 5. Examples of supportive scaffolding feedback, suggesting: (a) a missing object, *"you need a yellow gear here"*; (b) a shape, *"this object should be cut like this"*; and (c) a position, *"we need to move this to …"*

Strategic scaffolds. To make design tasks more manageable for young children, PrototypAR provides two types of strategic scaffolds (Figure 6a-b): first, PrototypAR highlights and limits the workspace to a particular area. Craft materials outside of the highlighted work area are ignored. Second, PrototypAR helps facilitate new design ideas by suggesting new structure attributes (*e.g.,* gear size) or behaviors (*e.g.,* focal length). This scaffold is intended to aid children in creating a set of prototypes for comparative experiment by letting them change one independent variable.

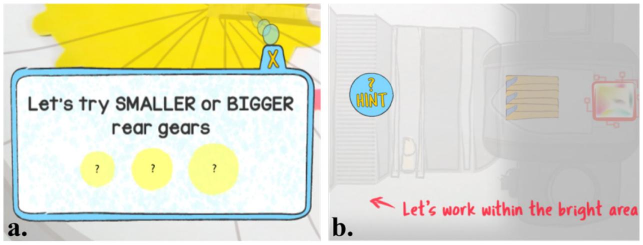

Figure 6. Examples of strategic scaffolds: (a) suggesting gears with different sizes; and (b) limiting the workspace to the area of the lens.

Procedural scaffolds. Procedural scaffolds help children use PrototypAR's prototyping interface as well as guide them through the iterative process of design and testing. For the first, the scaffolds remind children of paper colors for structure elements or illustrate how to design behavior labels (Figure 4) as needed. For the second, the system prompts the user to test the prototype when it is new, or asks for resuming design tasks after completing an experiment.

Virtual Simulations

At any point in the design process—from a partial prototype to a complete one—the user can test a digitized version of their work via *virtual simulations.* Simulations serve two purposes: first, to support the testing of a design to enhance understanding and discover potential flaws; second, to provide an experimental testbed to directly compare and analyze performance across prototype designs.

Towards these goals, we developed simulation support in both the AR design and the experiment modes. In the design mode, users can simulate individual components *in situ* via AR. This enables rapid testing of behavior, even at early stages of design. For example, the user can examine how the lens focus light rays at the focal point by watching an overlaid AR simulation. Users can then try different lens focal lengths in their workspace and observe the effect, which aids learning.

In the experiment mode, PrototypAR provides a simulation environment where users can test the function of prototypes and analyze results. While we custom built simulations for each application, the general approach is the same. Once the user enters the experiment mode, they are shown a *review panel* that displays images of their prototypes along with key design attributes (Figure 7a). The user can then select prototypes to test and begin the simulation. To facilitate controlled experimentation and reduce complexity, the review panel suggests clusters of prototypes that only differ in one design attribute (*e.g.,* rear gear size). After completing a simulation, an *analysis panel* organizes the results by shared independent variables so the user can easily analyze and compare results (Figure 7b).

101 mm
460 ms
Full Color
188 mm
49 mm

Figure 7. (a) The review panel shows a camera prototype along with its behavior variables. (b) The analysis panel shows pictures taken by two camera models that differ only by the focal length.

PrototypAR Implementation

PrototypAR is comprised of four sub-systems: (i) the *object recognition and model building* sub-system builds digital models from the paper prototypes; (ii) the *model assessment* engine evaluates the state of the digitized model; (iii) the *design manager* provides guidance and feedback to the user in the AR design mode; and (iv) the *experiment manager* handles the simulation environment in the experiment mode.

Object Recognition and Model Building Sub-System

The object recognizer analyzes the user's craft workspace and classifies paper elements as *structures* or *behaviors*. To avoid hand occlusion, PrototypAR's recognizer waits until there is no movement in the video stream for three seconds, obtained from informal experiments, before executing the recognition pipeline.

Recall that each structure element is pre-assigned a unique paper color. To recognize structures, we cluster the *hue* and *saturation* channels of the image into $K+1$ clusters, where K is equal to the total number of expected structures. We use *Gaussian Mixture Models* (GMMs) to train the K color models and cluster input pixels—a real-time method robust to camera noise [51,65]. To obtain shape information, we use

Please select three bikes to race (red, green, yellow)

Design #0
Gear Ratio 1.01
Front Gear 56 mm
Rear Gear 56 mm

Design #5
Gear Ratio 1.65
Front Gear 40 mm
Rear Gear 67 mm

Design #7
Gear Ratio 0.82
Front Gear 73 mm
Rear Gear 60 mm

Figure 8. The build-a-bike application. (a) The user creates a paper model consisting of gears (yellow for the rear, green for the front), chains (red), and pedal (blue); (b) the AR simulation shows animated components; (c) user selects three prototypes for experiment; (d) the virtual experiment simulates a race with the selected bikes; and (e) the simulation result shows the gear ratio of each bike to help analysis.

the 8-way flood fill algorithm [30,48] with the pixels in each color cluster to find the image blobs. Finally, the recognizer examines the connectivity between classified structures by examining spatial distances between objects. In all, the recognizer generates computational models of structures that include the object type, contour shape, position on the canvas, and connectivity to other objects.

For the behavior labels, we developed a *behavior recognizer,* which uses character recognition to determine the label type and an *input variable recognizer* that uses two approaches for recognizing the numeric and categorical data. To recognize the label type, we use the *Tesseract OCR* [60]. Once the label type is determined, PrototypAR examines the behavior variable. For numeric variables, PrototypAR uses blob detection to determine how much of the progress bar is filled in—the estimated fill portion is linearly mapped to a discrete value along a predefined range. For the categorical variables, PrototypAR divides the variable box into four quadrants and identifies the most saturated quadrant, which corresponds to a predefined behavior mode.

Model Assessment Engine

To assess the user's prototype, PrototypAR evaluates the computational model. The *model assessment engine* works by comparing the model to a pre-built baseline model. For structure, we evaluate the *shape, position, connectivity,* and *missing or redundant* structure elements. While some assessment algorithms are trivial (*e.g.* checking for the existence of a structure element), others are more complex. For example, to evaluate shape, we compare contours between the user's model and a baseline model using geometric distance. To ensure a robust comparison, the baseline model is scaled and transformed to minimize distance. If the distance is larger than a predefined threshold (determined via participatory design sessions), the assessment results in an *incorrect structure shape*. For behavior, we evaluate *missing* behaviors and *null* behavior variables, which require matching with the baseline model.

Design Manager

The *design manager* provides in-situ scaffolding feedback using the assessment results. When problems are found, the manager creates and visualizes supportive scaffolds. While *static scaffolds* render fixed visual content (*e.g.,* icons, text), *dynamic scaffolds* generate animations according to the user's model, often to show the user how to perform some action—for example, how to cut out a specific shape. To provide procedural scaffolds, the design manager monitors user interaction and records ongoing snapshots of a prototype and its corresponding digital model. When a digital model differs from the existing models, PrototypAR may suggest *testing* in the virtual simulation. For strategic scaffolds, the system dynamically dims and highlights part of the workspace to focus the user's attention. Finally, the design manager handles the *in situ* simulations of individual parts in the AR design mode.

Experiment Manager

The fourth and final sub-system, the *experiment manager*, controls the virtual simulations, including the *review panel,* the *simulation* environment itself, and the *analysis panel*. While the simulation environment and analysis panel need to be custom built for each application, the review panel provides a reusable architecture. Here, *PrototypAR* clusters similar prototypes together and helps organize experiments for prototypes that only differ in one *independent variable*. More specifically, given a pair of prototypes P_m and P_n, we calculate their *experimental distance D* as following:

$$\mathrm{D}(P_m, P_n) = \sum_{a_i \in A} d_{exp}(a_i, P_m, P_n)$$

$$d_{exp}(a, M, N) = \begin{cases} 1, & if\ design\ attribute\ a\ of\ M\ is\ diff\ from\ a\ of\ N \\ 0, & otherwise \end{cases}$$

Where A is a set of all design attributes. If $D(P_m, P_n) = 1$, we place both P_m and P_n in a cluster. The prototypes in a cluster can only differ by a single design attribute. After creating clusters through examining pairs, we merge clusters satisfying our conditions. Using this cluster information, the manager suggests a set of prototypes in the same cluster for experiment or comparative analysis.

Software Implementation

PrototypAR is implemented using *Unity3D* for creating the AR environment, *OpenCVSharp* [78] for computer vision, and *Parallel Extensions* in *.NET FX* for data parallelism.

Demo Applications

To demonstrate and evaluate PrototypAR, we developed three example applications: *build-a-bike*, *build-a-camera*, and *build-an-aquarium*—each allows children to design, build, and experiment with different types of complex systems from mechanics to optics to ecology.

Build-a-bike Application

In the *build-a-bike* application, children learn about bike gearing systems by modeling front gears, rear gears, pedals, and chains (Figure 8). This application explores gear ratio and chain drive system concepts. To build a bike, children first craft two gears, connect them via chains, and place a pedal at the center of the front gear. For behaviors, AR visualizations show the causal process of the pedal and the gears. For virtual experiments, the system simulates performances of gear designs in a bicycle race—depending on the gear ratio, one turn of the pedal can make the rear wheel turn less or more than one full cycle. Children can race up to three of their designs simultaneously.

Build-a-camera Application

In the *build-a-camera* application, children learn about camera optic systems by modeling lens, shutters, and sensors. The application emphasizes concepts of light focus and optical image sensing. To build a camera, children craft individual parts and then configure focal length, shutter speed, and sensor type via behavioral labels. AR visualizations show how light beams move through the parts and generate a picture (Figure 9). For virtual experiments, children can take pictures of scenes using their camera designs—*e.g.*, a city at night, a rainbow, and a safari. For the city scene, for example, children's camera design with a fast shutter speed results in an almost black picture. Setting a slower shutter, children can see the city in a resulting picture.

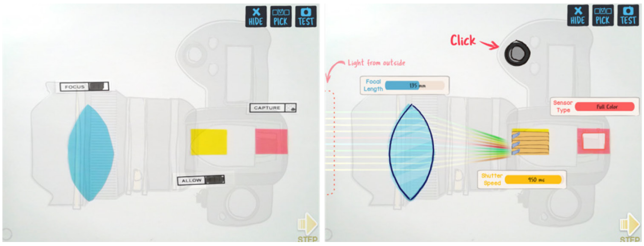

Figure 9. In *build-a-camera* application. (left) The model consists of lens (blue), shutter (yellow), and sensor (red). (right) The system visualizes the behaviors of individual components along with light rays.

Build-an-aquarium Application

In the build-an-aquarium application, children learn about aquatic ecology systems by modeling fish, sea plants, bacteria, and an air pump (inspired by [23]). This application emphasizes concepts of ecological balance and the nitrification process. To build an aquarium, children craft and distribute individual models over the canvas. For behaviors, AR visualizations show the causal process of air-pumps supplying oxygen, fish consuming oxygen, bacteria converting ammonia to nitrate, and plants consuming nitrate. For virtual experiments, the system simulates production and consumption of the chemicals by showing the current levels.

Figure 10. The *build-an-aquarium* application is shown: (left) the paper-based model; and (right) AR visualizations of individual objects and the simulated levels of chemicals.

USER STUDY

To examine how children interact with and use PrototypAR and to uncover opportunities and challenges for learning, we conducted four single-session evaluations with 21 children (ages 6-11; *M*=8.5; *SD*=1.6) at two local facilities—a children's museum and an after-school program. Based on our findings from the participatory design sessions, we recruited participants for each session based on age: (i) 10 younger children (ages 6-9) used the *build-a-bike* application in two sessions; and (ii) 11 older children (ages 9-11) used *build-a-camera* in the other two sessions. Our future work will address the *build-an-aquarium* application.

Method

All sessions followed the same general procedure but differed in length for administrative reasons: two sessions lasted 60 minutes and the others two lasted 90 minutes. Sessions began with a pre-activity questionnaire (5 minutes). Children were introduced to PrototypAR (5 or 10 minutes) and then used the system for 35 or 50 minutes. Finally, sessions concluded with a focus-group interview and post-activity questionnaire (15 or 25 minutes). Children worked in groups of two except one child who worked alone (*i.e.,* 5 groups used *build-a-bike* and 6 groups used *build-a-camera*). Each group had an adult facilitator who helped with PrototypAR and led the interviews.

After the introduction, children were given two tasks: first, to build at least one paper-based prototype that functioned properly in the simulator; and second, to complete a design challenge such as designing bike gears with certain performance or a camera to take pictures with a specified quality. The facilitators, if necessary, provided domain knowledge (*e.g.,* the meaning of gear ratio), prompted reflective discussions (*e.g.,* "*What do you think about the result?*"), and helped with resolving difficulties (*e.g., reading scaffolding texts for children*).

Data and Analysis

We collected pre- and post-activity questionnaires, photos and videos, focus-group interview, facilitator field notes, and system logs including interaction events and, crucially, prototype images—the latter enabled us to examine what each prototype looked like and how they changed over time. The questionnaires examined users' general experience with respect to engagement and usability using child-friendly Likert scale questions (based on [26]). The focus-group interviews asked open-ended questions to understand modeling and experiment experiences, children's learning, utility of the scaffolds, and design preferences.

To analyze the video data, we followed a peer-debriefing process [7,41]. We first formulated an initial coding scheme, which included the themes of engineering design process, how children interact with AR scaffolds, learning through construction and experimentation, and the role of peer support [52]. *Researcher A* coded a sample group's data and met with two researchers who were in the sessions to review the initial results and update the codebook by resolving disagreements, clarifying details, and generating new codes.

Researcher A then coded another random group's data and met with another researcher to review the results. After repeating this with another sample group's data, *Researcher A* coded the rest of the data. Finally, researchers synthesized findings including related quantitative data (*e.g.*, how many times children tested their models).

Findings

We describe patterns of design and iteration, interaction with system scaffolds, learning opportunities and challenges, collaboration, and engagement. For the Likert-scale questions, a rating of '5' indicates 'best.'

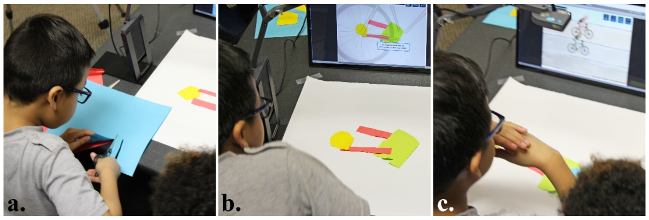

Figure 11. Children engaged in an iterative process of (a) making paper models, (b) evaluating the model through AR visualizations, and (c) experimenting with prototypes in the virtual simulation.

Design and iteration. We analyzed how children designed and evaluated prototypes with PrototypAR. System logs revealed that children approached design largely in two stages—first, a *bottom-up* step to build a complete model and then an *exploration* step to examine various forms of the complex system (Figure 12). We observed that, in early design stages, children focused on adding missing entities (*e.g.*, adding a chain), moving parts into the right places (*e.g.*, placing a gear at the center of the wheel), and refining shapes (*e.g.*, cutting a rectangular lens into an elliptical shape). Children progressively built parts until they had an initial model with properly sized, shaped, and placed components.

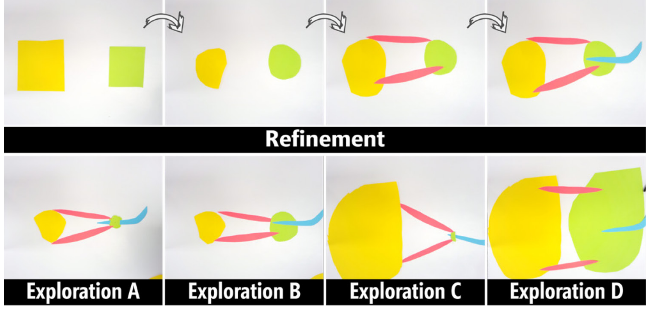

Figure 12. A group progressively built a complete bike model (above). Then, they created divergent prototypes for their experiments (below)

Once children built a complete design, they shifted their attention to explore a breadth of designs. Children replaced design entities (*e.g.*, replacing a front gear with a larger one, increasing a shutter speed) iteratively, often reusing existing paper pieces to quickly replicate a previous design. The system logs showed that groups created 7.8 distinct prototypes on average. The distinct prototypes exhibited different simulation results in the virtual experiment, which clarified how individual components function (*e.g.*, two camera models with fast or slow shutter speeds resulted in dark and bright pictures respectively). In response to the post-activity questionnaire item, "*I could see differences between prototypes in the virtual simulation*", all children except two selected '4' or '5' (*M*=4.6; *SD*=0.6). We also observed that children built "extreme" designs, which helped them explore and understand the design space. For example, in the *build-a-bike* application, 3 of 5 groups created both giant and tiny gears. One child stated, "*It's going to be funny! It's going to be funny!*" as they made a giant gear.

In both stages of design, we noted that the AR visualization and in-situ experiment feedback prompted children to try new design ideas. First, children identified design issues by observing how changes in individual components affected the simulation. For example, a child realized the gears in his prototype were not rotating due to missing chains; he said, "*We need to connect two gears...otherwise it wouldn't move.*" This example demonstrates how PrototypAR's just-in-time feedback prompted children to realize that their system was missing a component (*i.e.,* chains) and was therefore incomplete. In addition, the interactive simulation results prompted children to reflect on their prototype designs as a whole. For example, in the *build-a-bike* experiment, one child suggested increasing a front gear after watching a bike with a larger rear gear lose a race saying, "*I think the front [gear] has to be big. [The rear gear] has to be small*". Similarly, in the *build-a-camera* experiment, one child suggested changing a shutter speed after seeing a dark picture taken by a camera prototype saying, *"let's try a full [shutter speed]"* On the post-activity questionnaire item, "*I think the Test (virtual experiment) was helpful*", children appreciated the usefulness of the virtual experiment; 15 of 21 selected '4' or '5' (*M*=4.0; *SD*=1.2). In the interview, a child affirmed it stating, *"It helped a lot, if [there was] no test button, we couldn't know how good the camera is."*

Interactions with scaffolds. Children used and reacted to the three scaffold types differently. First, children used the supportive scaffolds, which provided design feedback, to evaluate individual models but used them less often as they gained experience. In early design phases, we observed that children made use of supportive scaffolds almost whenever one was available. They chose to open a *Hint*, read the feedback dialog, watched animations of design suggestions, and discussed the ideas therein. When asked if the scaffold was helpful on the post-activity questionnaire, 18 of 21 participants selected '4' or '5' (*M*=4.5; *SD*=0.8). A child stated, "*It helped you make the bike.*" However, we found from video data that children did not fully follow the design suggestions; rather, they used their own ideas or interests for designs. For example, two groups created and tested rectangular gears while the scaffold suggested a circular shape. In the later phases of design, children became less likely to use *Hint* scaffolds. From the system logs, we found that 76% of Hint usage, on average (*SD*=14%), occurred in the first half of the design process.

Strategic scaffolding that illuminated and constrained the current work area (*e.g.,* highlighting the area around the lens in the *build-a-camera* application) seemed to help children *divide and conquer* the complexity of a design. For example, from the system logs, we found that all groups successively created at least three different designs for a specific part when the workspace was limited. After iterating on a part, children repeatedly switched the workspace to the other part until they had a full-fledged prototype. In contrast, children did not always seem to follow the strategic scaffold that actively prompted them to explore specific design attributes (*e.g.,* a dialog suggests increasing or decreasing a front gear size). From system logs, we found that children had already started modifying these attributes before receiving the suggestion or simply did not follow PrototypAR suggestions even after reading them.

Collaboration. We analyzed how the tangible approach supported communicating ideas [63], sharing control [76], and concurrent interaction [20]. Though children were not assigned specific roles during the activity, from the video data, we observed a set of collaborative behaviors including splitting design tasks, discussing design ideas, and sharing observations. For example, Emma and Noah were working together on designing a camera shutter. Noah read design feedback about the shape and clarified it talking to Emma, "*Just make it like a small square. It doesn't have to be like same size.*" Later, Emma wondered about the level of the shutter speed, asking "*Should we make it full?*" Noah nodded saying, "*Full! Full!*" Finally, in the virtual experiment, Noah compared two pictures taken by different camera models and explained how the focal lengths influenced them stating, *"This is zoomed-in and this is zoomed-out."*

However, we also observed that children had difficulties managing conflicts in their design ideas and manipulating a shared virtual interface. For example, when Ava and Liam were making a bike prototype, Liam suddenly cut an existing front pedal without discussion, and Ava got annoyed shouting, "*What are you doing*!?" In another example, Ethan and Jacob were selecting bike prototypes to simulate. When Ethan was selecting prototypes, Jacob suddenly stopped Ethan saying "*I will do this,*" complaining, "*You did last time. Can I do it this time?*" These conflicts were resolved by a facilitator.

Content learning. We examined how using PrototypAR contributed to children's understanding of complex systems. These results should be considered preliminary given the small sample size. During the activity and the group interview, 10 of 11 groups reported that they learned about what objects exist in a complex system and how they behave. For example, a child whose group succeeded in creating a complete camera model after 11 iterations stated, "*We learned three different parts of camera.*" The other child in the same group added, "*We learned how to make it [the lens] focus…learned [the] shutter allows light to pass or not.*" Another child—who tested different focal lengths and observed the resulting phenomena in the AR visualizations—reported that he learned how a lens controls light stating, "*Lens makes the light focus at one place.*"

While all the groups reported their findings about how system components influence the system's function, we found that their understanding could be incorrect or partial. From verbal observations they made while tinkering with the simulations and in their responses to the interview question "*what did you learn?,*" children shared accurate conceptions of how individual parts contribute to a system's function including: "*Bigger rear gear does not make the bike faster*" and "*If we don't put the shutter, it's [the picture is] just all bright.*" We found that 2 of 5 groups who used *build-a-bike* demonstrated misunderstandings such as "*If green [front] and yellow [rear] gears are small, it makes the bike slower.*" And, 4 of 6 groups who used *build-a-camera* ended up with partial understandings about the system—*e.g.,* a group could not grasp how the shutter works but demonstrated understandings about the lens and the sensor. We return to these misconceptions in the Discussion.

Experimentation challenges. Related to the above, we observed two primary challenges children had in conducting experiments with PrototypAR: designing experiments and analyzing observations. To understand the relationships between design attributes and a system's function, it is critical to design and conduct comparative experiments—testing a set of prototypes that have different attributes for a single independent variable. Though PrototypAR automatically suggests a selection of appropriate prototypes to compare, we found that children often selected designs that looked most different or even, seemingly, at random. This made it difficult for children to make accurate claims from reviewing the experiment results. For example, in the *build-a-bike* application, a group ran experiments with a big prototype having two big gears and prototypes having gears of different sizes, and concluded with the misconception, "*If gears are same size, the bike goes faster.*"

We also observed that children had difficulties analyzing the simulation results. Even in cases with well-executed experiments, children often could not explain why they got the results or drew inaccurate conclusions. For example, a group tested a camera with a fast shutter speed to take a picture of a dark scene that actually requires a slow shutter speed. When the simulation resulted in very dark photos, they could not reason why this happened and became disengaged after several tries. A child in the group commented in the later interview, *"[it was] difficult to be [the] color you wanted."* The group even thought it was a system malfunction, asking a facilitator to fix the problem.

Engagement. The majority of participants reported having fun with PrototypAR; 16 of 21 children responded '4' or '5' (*M*=3.8; *SD*=1.6) to the post-activity questionnaire item, *"I had fun using PrototypAR."* In group interviews, children liked using craft materials (*e.g., "Using different materials and colors"*), making a creative or extreme design (*e.g., "huge gear"*), AR visualizations (*e.g., "Cool effect on white paper"*), and virtual simulation for testing (*e.g., "To see what pictures would look like"*). However, four participants had a negative experience. One participant commented that the visual gaps between real objects and virtual objects made it less interesting: *"We got to have this gigantic [real wheel], but we have this tiny [virtual wheel]."* We also found that repeatedly making the same system (*e.g., "Making a lot of bikes"*) and constraining design (*e.g., "It wasn't so exciting, I had to follow lots of rules"*) made the process seem tedious.

DISCUSSION

We studied two PrototypAR applications using a single-session study design. While this is appropriate for our exploratory goal of studying user interaction, investigating opportunities and challenges, and drawing design implications, the study is insufficient for examining learning or long-term engagement. Our findings show that a mixed reality approach—accompanied with scaffolding—can allow children to engage with modeling and experimentation of complex systems. This suggests that complex systems learning is approachable for young children given appropriate learner-centered tools and environments, extending Danish *at el.*'s findings [15].

Learner-centered approach. With PrototypAR, we envisioned a learner-centered environment [27] where children can address their unique interests and deepen understanding. Specifically, we posited that children can learn about different aspects of complex systems by constructing the *structure* of a system model, observing AR simulations of component *behaviors*, and comparing the *functions* of their different designs in the virtual experiment. Indeed, the groups were able to learn different aspects of a complex system from the same activity. For example, in the *build-a-camera* application, one group reported learning about how the focal length affects the zoom-level of a picture while another learned about the shutter affects the brightness of a picture. Children enjoyed having this level of control in their design and experimentation process. This tendency resulted in positive outcomes such as engagement with design iterations and unexpected findings (*e.g.*, a child was surprised to see bigger chains did not affect the bike speed). But, it also limits opportunities to examine all the parts of a complex system and develop understanding about how the system works as a whole, which often led to partial understandings. Future work should consider scaffolds that can support iterative expansion of children's component-level focus while highlighting comprehensive interrelationships and functions of these components.

Tinkering vs. structured scaffolding. Constructionist learning environments that support playful exploration can afford children's serendipitous opportunities for "ah-ha" moments, yield options for experimental comparison [62], are more aligned to authentic science inquiry as practiced by professionals [9], and may promote intellectual risk taking, a key for science learning [4]. Likewise, our findings suggest that free-form prototyping promoted children's engagement and encouraged personal, interest-driven experimentation. However, their prototypes did not always lead to systems-level understanding or accurate mental models. Their enjoyment with testing the extreme bounds of a design ("huge gears!") hinted at a nascent awareness of design constraints, but lacked a systematic approach, such as controlling for variables. Moreover, the children's eagerness to create silly, random designs often precluded them from taking up the system's scaffolded suggestions, which led to misconceptions. These findings affirm the need to balance learners' free-form play with structured guidance for inquiry [10]. Future designs should consider how scaffolds can respond and adapt to children's own ideas, in minimalist but directed ways that guide their efforts to design and execute systematic modes of inquiry. Because children often ignore or feel constrained by lock-step scaffolds that limit their design freedom, future work should also consider interactive design features that prompt learners to reflect upon their ideas and modify them iteratively rather than randomly.

Tangible limitations. Our findings suggest that PrototypAR's tangible prototyping interface lowers entry barriers to modeling complex systems and helps children understand visual and spatial aspects of complex systems. However, our current system does not yet support more complex models that may involve layered, occluding structures, large numbers of interacting components, or ways to represent abstract processes [34,38]. To address these limitations, future work should explore hybrid approaches of combining physical and virtual interfaces, extending the current 2D design space to 3D, and adding auxiliary input modalities (*e.g.,* voice or embodied interaction).

AR design environment. While prior work has explored AR modeling systems for adults or high school students [39,58], our work demonstrates the benefits of AR for elementary-level children to access domain knowledge via supportive scaffolds, deal with design complexity in guidance of strategic scaffolds, and draw design ideas from reflections on AR visualization of models. However, our current AR approach limits immersion. The user interface is distributed across the physical desk and the screen, which can negatively affect usability. For example, we observed that some children tried to select virtual menus on the screen by tapping the canvas. Future work should explore other AR techniques (*e.g.,* projection display) to integrate the physical and virtual.

CONCLUSION

Our paper introduces PrototypAR, an AR system to support complex systems learning through iterative craft modeling, AR-based scaffolding, and virtual experiments. Through an iterative design process involving children participants, we designed and developed the *smart desk* system along with three applications. The evaluation of two PrototypAR applications helps understand how children iteratively design and test complex systems models, their interaction with AR-realized scaffolds, and challenges in learning through cycles of design and experimentation. Our findings suggest children's engagement with complex systems learning, refinement and exploration of designs through iterations, and opportunities and challenges of scaffolds for design and experimentation.

SELECTION AND PARTICIPATION OF CHILDREN

We recruited children from two local facilities and an on-campus design group. We initially had a meeting with the facility managers to explain our research and distribute the informed consent describing research procedures and data acquisition to parents. The children participated in the study if their parents signed the consent.

ACKNOWLEDGEMENTS

We thank the children and researchers in the KIDSteam program at the University of Maryland, College Park. This work was supported by NSF Grant IIS-441184 and IIS-1652339.

REFERENCES

[1] Michelle Annett, Tovi Grossman, Daniel Wigdor, and George Fitzmaurice. 2015. MoveableMaker: Facilitating the Design, Generation, and Assembly of Moveable Papercraft. *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*: 565–574. https://doi.org/10.1145/2807442.2807483

[2] Orit Ben Zvi Assaraf and Nir Orion. 2005. Development of system thinking skills in the context of earth system education. *Journal of Research in Science Teaching* 42, 5: 518–560. https://doi.org/10.1002/tea.20061

[3] Roger Azevedo, John T. Guthrie, and Diane Seibert. 2004. The Role of Self-Regulated Learning in Fostering Students' Conceptual Understanding of Complex Systems with Hypermedia. *Journal of Educational Computing Research* 30, 1–2: 87–111. https://doi.org/10.2190/DVWX-GM1T-6THQ-5WC7

[4] Ronald A Beghetto. 2009. Correlates of intellectual risk taking in elementary school science. *Journal of Research in Science Teaching: The Official Journal of the National Association for Research in Science Teaching* 46, 2: 210–223.

[5] Amanda M. Bell. 2015. Learning complex systems with story-building in scratch. In *Proceedings of the 14th International Conference on Interaction Design and Children*, 307–310. https://doi.org/10.1145/2771839.2771903

[6] Lars Bollen and Wouter R. Van Joolingen. 2013. SimSketch: Multiagent simulations based on learner-created sketches for early science education. *IEEE Transactions on Learning Technologies* 6, 3: 208–216. https://doi.org/10.1109/TLT.2013.9

[7] Virginia Braun and Victoria Clarke. 2006. Using thematic analysis in psychology. *Qualitative Research in Psychology* 3, 2: 77–101.

[8] Kursat Cagiltay. 2006. Scaffolding strategies in electronic performance support systems: Types and challenges. *Innovations in Education and Teaching International* 43, 1: 93–103. https://doi.org/10.1080/14703290500467673

[9] Clark A. Chinn and Betina A. Malhotra. 2002. Epistemologically authentic inquiry in schools: A theoretical framework for evaluating inquiry tasks. *Science Education* 86, 2: 175–218. https://doi.org/10.1002/sce.10001

[10] Tamara Clegg, Elizabeth Bonsignore, Jason Yip, Helene Gelderblom, Alex Kuhn, Tobin Valenstein, Becky Lewittes, and Allison Druin. 2012. Technology for promoting scientific practice and personal meaning in life-relevant learning. In *Proceedings of the 11th International Conference on Interaction Design and Children*, 152–161.

[11] Tamara Clegg, Leyla Norooz, Seokbin Kang, Virginia Byrne, Monica Katzen, Rafael Velez, Angelisa Plane, Vanessa Oguamanam, Thomas Outing, Jason Yip, and others. 2017. Live physiological sensing and visualization ecosystems: an activity theory analysis. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, 2029–2041.

[12] Allan Collins, John Seely Brown, and Susan E Newman. 1989. Cognitive apprenticeship: Teaching the crafts of reading, writing, and mathematics. *Knowing, learning, and instruction: Essays in honor of Robert Glaser* 18: 32–42.

[13] National Research Council and others. 2000. *How people learn: Brain, mind, experience, and school: Expanded edition*. National Academies Press.

[14] Allen Cypher and David Canfield Smith. 1995. KidSim: End user programming of simulation. *Human Factors in Computing Systems (CHI)*: 27–34. https://doi.org/10.1145/223904.223908

[15] J A Danish, K Peppler, and D Phelps. 2010. BeeSign: Designing to support mediated group inquiry of complex science by early elementary students. In *Proceedings of the 9th International Conference on Interaction Design and Children*, 182–185. https://doi.org/10.1145/1810543.1810566

[16] Daniel C. Edelson, Douglas N. Gordin, and Roy D. Pea. 1999. Addressing the Challenges of Inquiry-Based Learning Through Technology and Curriculum Design. *Journal of the Learning Sciences* 8, 3–4: 391–450. https://doi.org/10.1080/10508406.1999.9672075

[17] Lyn D. English. 2016. STEM education K-12: perspectives on integration. *International Journal of STEM Education* 3, 1: 3. https://doi.org/10.1186/s40594-016-0036-1

[18] Maria Evagorou, Kostas Korfiatis, Christiana Nicolaou, and Costas Constantinou. 2009. An investigation of the potential of interactive simulations for developing system thinking skills in elementary school: A case study with fifth-graders and sixth-

graders. *International Journal of Science Education* 31, 5: 655–674. https://doi.org/10.1080/09500690701749313

[19] Jerry Alan Fails, Mona Leigh Guha, Allison Druin, and others. 2013. Methods and techniques for involving children in the design of new technology for children. *Foundations and Trends®in Human--Computer Interaction* 6, 2: 85–166.

[20] Ylva Fernaeus and Jakob Tholander. 2006. Finding design qualities in a tangible programming space. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, 447–456. https://doi.org/10.1145/1124772.1124839

[21] Rochel Gelman and Kimberly Brenneman. 2004. Science learning pathways for young children. *Early Childhood Research Quarterly* 19, 1: 150–158. https://doi.org/10.1016/j.ecresq.2004.01.009

[22] Ashok K. Goel, Spencer Rugaber, and Swaroop Vattam. 2009. Structure, behavior, and function of complex systems: The structure, behavior, and function modeling language. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 23, 01: 23. https://doi.org/10.1017/S0890060409000080

[23] Ashok K Goel, Swaroop S Vattam, Spencer Rugaber, David Joyner, Cindy E Hmelo-silver, Rebecca Jordan, Sameer Honwad, Steven Gray, and Suparna Sinha. 2010. Learning Functional and Causal Abstractions of Classroom Aquaria. *Proceedings of the Annual Meeting of the Cognitive Science Society* 32, 32.

[24] Mona Leigh Guha, Allison Druin, and Jerry Alan Fails. 2013. Cooperative Inquiry revisited: Reflections of the past and guidelines for the future of intergenerational co-design. *International Journal of Child-Computer Interaction* 1, 1: 14–23. https://doi.org/10.1016/j.ijcci.2012.08.003

[25] Mark Guzdial. 1994. Software-realized scaffolding to facilitate programming for science learning. *Interactive Learning Environments* 4, 1: 1–44.

[26] Lynne Hall, Colette Hume, and Sarah Tazzyman. 2016. Five degrees of happiness: effective smiley face Likert Scales for evaluating with children. In *Proceedings of the The 15th International Conference on Interaction Design and Children*, 311–321.

[27] Michael J Hannafin and Susan M Land. 1997. The foundations and assumptions of technology-enhanced student- centered learning environments. *Instructional Science* 25, 3: 167–202. https://doi.org/10.1023/A:1002997414652

[28] 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*, 299–308. https://doi.org/10.1145/1166253.1166300

[29] Eiji Hayashi, Martina Rau, Zhe Han Neo, Nastasha Tan, Sriram Ramasubramanian, and Eric Paulos. 2012. TimeBlocks: “Mom, can I have another block of time?” In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems*, 1713–1716. https://doi.org/10.1145/2207676.2208299

[30] Paul S Heckbert. 1990. A seed fill algorithm. In *Graphics gems*, 275–277.

[31] Geoffrey Hinchliffe. 2002. Situating Skills. *Journal of Philosophy of Education* 36, 2: 187–205. https://doi.org/10.1111/1467-9752.00269

[32] Cindy E. Hmelo-Silver, Ravit Golan Duncan, and Clark a. Chinn. 2007. Scaffolding and achievement in problem-based and inquiry learning: A response to Kirschner, Sweller, and Clark (2006). *Educational Psychologist* 42, 2: 99–107. https://doi.org/10.1080/00461520701263368

[33] Cindy E. Hmelo-Silver, Surabhi Marathe, and Lei Liu. 2007. Fish swim, rocks sit, and lungs breathe: Expert-novice understanding of complex systems. *Journal of the Learning Sciences* 16, 3: 307–331. https://doi.org/10.1080/10508400701413401

[34] Cindy E. Hmelo, Douglas L. Holton, and Janet L. Kolodner. 2000. Designing to Learn About Complex Systems. *Journal of the Learning Sciences* 9, 3: 247–298. https://doi.org/10.1207/S15327809JLS0903

[35] Juan Pablo Hourcade. 2007. Interaction Design and Children. *Foundations and Trends® in Human-Computer Interaction* 1, 4: 277–392. https://doi.org/10.1561/1100000006

[36] Scott E. Hudson and Jennifer Mankoff. 2006. Rapid construction of functioning physical interfaces from cardboard, thumbtacks, tin foil and masking tape. In *Proceedings of the 19th annual ACM symposium on User interface software and technology*, 289–298. https://doi.org/10.1145/1166253.1166299

[37] Maria Blanca Ibanez, Diego Villar, Carlos Delgado-kloos, and Senior Member. 2016. Support for Augmented Reality Simulation Systems : The Effects of Scaffolding on Learning Outcomes and Behavior Patterns n. 9, 1: 46–56.

[38] Michael J. Jacobson and Uri Wilensky. 2006. Complex Systems in Education: Scientific and Educational Importance and Implications for the Learning Sciences. *Journal of the Learning Sciences* 15, 1: 11–34. https://doi.org/http://dx.doi.org/10.1207/s15327809jls1501_4

[39] Hannes Kaufmann and Dieter Schmalstieg. 2003. Mathematics and geometry education with

collaborative augmented reality. *Computers and Graphics (Pergamon)* 27, 3: 339–345. https://doi.org/10.1016/S0097-8493(03)00028-1

[40] Scott R. Klemmer, Anoop K. Sinha, Jack Chen, James a. Landay, Nadeem Aboobaker, and Annie Wang. 2000. Suede: a Wizard of Oz prototyping tool for speech user interfaces. In *Proceedings of the 13th annual ACM symposium on User interface software and technology*, 1–10. https://doi.org/10.1145/354401.354406

[41] Yvonna S Lincoln. 1985. Naturalistic inquiry. *The Blackwell Encyclopedia of Sociology*.

[42] Lei Liu and Cindy E. Hmelo-Silver. 2009. Promoting complex systems learning through the use of conceptual representations in hypermedia. *Journal of Research in Science Teaching* 46, 9: 1023–1040. https://doi.org/10.1002/tea.20297

[43] Joanne Lo, Cesar Torres, Isabel Yang, Jasper O'Leary, Danny Kaufman, Wilmot Li, Mira Dontcheva, and Eric Paulos. 2016. Aesthetic Electronics: Designing, Sketching, and Fabricating Circuits Through Digital Exploration. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, 665–676. https://doi.org/10.1145/2984511.2984579

[44] Michael R. Marner and Bruce H. Thomas. 2010. Augmented foam sculpting for capturing 3D models. In *IEEE Symposium on 3D User Interfaces 2010, Proceedings*, 63–70. https://doi.org/10.1109/3DUI.2010.5444720

[45] Paul Marshall. 2007. Do tangible interfaces enhance learning? In *Proceedings of the 1st international conference on Tangible and embedded interaction* (TEI '07), 163–170. https://doi.org/10.1145/1226969.1227004

[46] Katherine L. McNeill, David J. Lizotte, Joseph Krajcik, and Ronald W. Marx. 2006. Supporting students' construction of scientific explanations by fading scaffolds in instructional materials. *Journal of the Learning Sciences* 15, 2: 153–191. https://doi.org/10.1207/s15327809jls1502_1

[47] NGSS Lead States. 2013. Next Generation Science Standards. *Achieve, Inc. on behalf of the twenty-six states and partners that collaborated on the NGSS*, November: 1–103. https://doi.org/10.17226/18290

[48] Theodosios Pavlidis. 2012. *Algorithms for graphics and image processing*. Springer Science & Business Media.

[49] David E. Penner. 2000. Explaining systems: Investigating middle school students' understanding of emergent phenomena. *Journal of Research in Science Teaching* 37, 8: 784–806. https://doi.org/10.1002/1098-2736(200010)37:8<784::AID-TEA3>3.0.CO;2-E

[50] Kylie Peppler, Joshua Danish, Benjamin Zaitlen, Diane Glosson, Alexander Jacobs, and David Phelps. 2010. BeeSim: leveraging wearable computers in participatory simulations with young children. *Proceedings of the 9th International Conference on Interaction Design and Children*: 246–249. https://doi.org/10.1145/1810543.1810582

[51] Haim Permuter, Joseph Francos, and Ian Jermyn. 2006. A study of Gaussian mixture models of color and texture features for image classification and segmentation. *Pattern Recognition* 39, 4: 695–706. https://doi.org/10.1016/j.patcog.2005.10.028

[52] Sadhana Puntambekar and Roland Hübscher. 2005. Tools for Scaffolding Students in a Complex Learning Environment: What Have We Gained and What Have We Missed? *Educational Psychologist* 40, 1: 1–12.

[53] Chris Quintana, Brian Reiser, Elizabeth Davis, Joseph Krajcik, Eric Fretz, Ravit Golan Duncan, Eleni Kyza, Daniel Edelson, and Elliot Soloway. 2004. A Scaffolding Design Framework for Software to Support Science Inquiry. *The Journal of the Learning Sciences* 13, 3: 337–386. https://doi.org/10.1207/s15327809jls1303_4

[54] Iulian Radu and Blair Macintyre. 2012. Using Children ' s Developmental Psychology to Guide Augmented-Reality Design and Usability. *IEEE International Symposium on Mixed and Augmented Reality 2012*: 227–236. https://doi.org/10.1109/ISMAR.2012.6402561

[55] Alexander Repenning, Andri Ioannidou, and Jonathan Phillips. 1999. Collaborative use & design of interactive simulations. *Proceedings of the 1999 conference on Computer support for collaborative learning CSCL 99*: 59–es. https://doi.org/10.3115/1150240.1150299

[56] M. Resnick. 1996. StarLogo: An environment for decentralized modeling and decentralized thinking. *Conference Companion on Human Factors in Computing Systems (CHI '96)*: 11–12. https://doi.org/10.1145/257089.257095

[57] Mitchel Resnick. 1996. Beyond the Centralized Mindset. *Journal of the Learning Sciences* 5, 1: 1–22. https://doi.org/10.1207/s15327809jls0501_1

[58] Teresa Restivo, Fátima Chouzal, José Rodrigues, Paulo Menezes, and J. Bernardino Lopes. 2014. Augmented reality to improve STEM motivation. *IEEE Global Engineering Education Conference, EDUCON*, April: 803–806. https://doi.org/10.1109/EDUCON.2014.6826187

[59] Bertrand Schneider, Patrick Jermann, Guillaume

Zufferey, and Pierre Dillenbourg. 2011. Benefits of a tangible interface for collaborative learning and interaction. *IEEE Transactions on Learning Technologies* 4, 3: 222–232. https://doi.org/10.1109/TLT.2010.36

[60] Ray Smith, Ray Smith, and Google Inc. 2007. An overview of the Tesseract OCR Engine. In *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, 629--633. https://doi.org/https://doi.org/10.1109/ICDAR.2007.4376991

[61] Hyunyoung Song, François Guimbretière, Chang Hu, and Hod Lipson. 2006. ModelCraft: capturing freehand annotations and edits on physical 3D models. In *Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology*, 13–22. https://doi.org/10.1145/1166253.1166258

[62] Technically Speaking. 2008. Advancing Engineering Education in P-12 Classrooms in P-12 Classrooms. July.

[63] Hideyuki Suzuki and Hiroshi Kato. 1995. Interaction-level support for collaborative learning: AlgoBlock—an open programming language. *The first international conference on Computer support for collaborative learning - CSCL '95*, 349–355. https://doi.org/10.3115/222020.222828

[64] Seth Tisue and Uri Wilensky. 2004. Netlogo: A simple environment for modeling complexity. *Conference on Complex Systems*: 1–10. https://doi.org/10.1109/ICVD.2004.1261037

[65] Real-time Tracking and Eric Thul. 2007. Adaptive Background Mixture Models for. 00, c.

[66] Swaroop S Vattam, Ashok K Goel, Spencer Rugaber, Cindy E Hmelo-silver, Steven Gray, and Suparna Sinha. 2011. Understanding Complex Natural Systems by Articulating Structure-Behavior- Function Models. *Educational Technology & Society , Special Issue on Creative Design* 14, 1: 66–81.

[67] Lev Semenovich Vygotsky, Robert W Rieber, and Marie J Hall. 1998. *The collected works of LS Vygotsky, Vol. 5: Child psychology.* Plenum Press.

[68] Uri Wilensky and New Orleans. 2002. Participatory Simulations : Envisioning the networked classroom as a way to support systems learning for all Presented at the Annual meeting of the American Educational Research. *Network.*

[69] Uri Wilensky and Walter Stroup. 1999. Learning Through Participatory Simulations: Network-based Design for Systems Learning in Classrooms. In *Proceedings of the 1999 Conference on Computer Support for Collaborative Learning* (CSCL '99). Retrieved from http://dl.acm.org/citation.cfm?id=1150240.1150320

[70] M Wilkerson-Jerde, B Gravel, and C Macrander. 2013. SiMSAM: An integrated toolkit to bridge student, scientific, and mathematical ideas using computational media. In *Proceedings of the international conference of computer supported collaborative learning (CSCL 2013)*, 379–381.

[71] Michelle H. Wilkerson-Jerde, Brian E. Gravel, and Christopher A. Macrander. 2015. Exploring Shifts in Middle School Learners' Modeling Activity While Generating Drawings, Animations, and Computational Simulations of Molecular Diffusion. *Journal of Science Education and Technology* 24, 2–3: 396–415. https://doi.org/10.1007/s10956-014-9497-5

[72] Michelle Wilkerson-Jerde, Aditi Wagh, and Uri Wilensky. 2015. Balancing Curricular and Pedagogical Needs in Computational Construction Kits: Lessons From the DeltaTick Project. *Science Education* 99, 3: 465–499. https://doi.org/10.1002/sce.21157

[73] Hsin-kai Wu, Silvia Wen-yu Lee, Hsin-yi Chang, and Jyh-chong Liang. 2013. Current status, opportunities and challenges of augmented reality in education. *Computers & Education* 62: 41–49. https://doi.org/10.1016/j.compedu.2012.10.024

[74] Z. C. Zacharia. 2007. Comparing and combining real and virtual experimentation: An effort to enhance students' conceptual understanding of electric circuits. *Journal of Computer Assisted Learning* 23, 2: 120–132. https://doi.org/10.1111/j.1365-2729.2006.00215.x

[75] Zacharias Zacharia and O Roger Anderson. 2003. The effects of an interactive computer-based simulation prior to performing a laboratory inquiry-based experiment on students' conceptual understanding of physics. *American Journal of Physics* 71, 6: 618–629.

[76] Oren Zuckerman, Saeed Arida, and Mitchel Resnick. 2005. Extending tangible interfaces for education. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '05* (CHI '05), 859. https://doi.org/10.1145/1054972.1055093

[77] Oren Zuckerman, T Grotzer, and K Leahy. 2006. Flow blocks as a conceptual bridge between understanding the structure and behavior of a complex causal system. In *Proceedings of the 7th international conference on Learning sciences*, 880–886.

[78] 2018. OpenCVSharp. Retrieved from https://github.com/shimat/opencvsharp