Investigating How Online Help and Learning Resources Support Children's Use of 3D Design Software

Nathaniel Hudson[*][†]**, Benjamin Lafreniere**[*]**, Parmit K. Chilana**[‡]**, Tovi Grossman**[*]

[*]Autodesk Research
Toronto, Ontario, Canada
{*ben.lafreniere*, *tovi.grossman*}@autodesk.com

[†]Ross Video
Ottawa, ON, Canada
me@nathanielh.com

[‡]Simon Fraser University
Burnaby, BC, Canada
pchilana@cs.sfu.ca

ABSTRACT

3D design software is increasingly available to children through libraries, maker spaces, and for free on the web. This unprecedented availability has the potential to unleash children's creativity in cutting edge domains, but is limited by the steep learning curve of the software. Unfortunately, there is little past work studying the breakdowns faced by children in this domain—most past work has focused on adults in professional settings. In this paper, we present a study of online learning resources and help-seeking strategies available to children starting out with 3D design software. We find that children face a range of challenges when trying to learn 3D design independently—tutorials present instructions at a granularity that leads to overlooked and incorrectly-performed actions, and online help-seeking is largely ineffective due to challenges with query formulation and evaluating found information. Based on our findings, we recommend design directions for next-generation help and learning systems tailored to children.

Author Keywords

Software learning; child-computer interaction; online help.

ACM Classification Keywords

H.5.m. Information interfaces and presentation: Misc.

INTRODUCTION

The last few years have seen the rising popularity of simplified and free versions of consumer-oriented 3D design tools that can be easily accessed by non-professional users [53]. As part of this trend, children are a growing user group for 3D design software [72]. These tools offer the potential to unleash children's creativity in cutting edge domains such as video game design [74], animation [75], and digital fabrication [22]. Moreover, building an interest in 3D design at an early age may inspire children to pursue engineering or a related scientific field later in life [68].

Although 3D design tools have become more easily accessible, and offer great potential, they can still be difficult to learn and use, and getting started remains a significant barrier, even for adults [35, 41]. Since users have long abandoned static manuals and documentation [64], HCI experts and product designers have developed alternative learning and help techniques that offer more targeted and interactive assistance. For example, with the web emerging as the dominant platform for exchanging software-related experiences, HCI research has investigated novel ways of integrating web-based resources into design tools, including Q&A features [54], step-by-step tutorials [30, 49, 50], videos [55, 61], contextual help [26, 46], and search interfaces [49], to name a few.

While these innovations in software help and learning are promising, most of them have been designed based on formative studies with educated adults, typically in professional or formal educational contexts [35, 66], and we know little about *children's* experiences and struggles with using common online instructional materials (e.g., in-app, video, and web-based tutorials) and help-seeking resources (e.g., web search, forums) when learning new software. Although there is a rich body of research on children's help-seeking behaviors in classrooms [33, 58] and online learning environments [2, 3], the focus has mostly been on scholastic tasks, such as learning mathematics. Understanding the breakdowns faced by children, and the effectiveness of currently-available help and learning resources is important to inform the design of a new generation of help systems and learning resources. These resources may support not only children, but also adults in using 3D design tools and feature-rich software more broadly.

In this paper, we investigate the barriers faced by children when learning 3D design software, by evaluating the effectiveness of common help and learning resources that are available to children on the web. Our investigation addresses two main research questions: (1) What challenges do children face in using instructional tutorials when learning 3D design software? and, (2) To what extent do common online help-seeking behaviors, such as use of web search and online community help, support children in learning and using 3D design tools?

To address these questions, we conducted an in-lab study with children, ages 10–15, following tutorials in 3D design software in an individual setting. We intentionally prescribed different types of learning materials—including video, web-based, and in-application tutorials—and different help-seeking strategies to gain insights into the most prevalent online

resources available to individual learners. The design of this study was informed by interviews with eight facilitators of 3D design workshops that shed light on current software tools and tasks used for teaching 3D design to children.

Our key findings suggest that children can face a range of challenges when trying to seek help and learn 3D design tools using online instructional materials. For example, when attempting to follow web-based tutorials, the children in our study often overlooked actions or performed them incorrectly, and experienced challenges with locating the UI elements required to carry out tutorial steps. In addition, we found that help-seeking practices such as the use of web search and community forums—resources commonly used by adults—were also ineffective for children. In particular, children struggled with evaluating the value and trustworthiness of search results and community-created content, and with formulating queries to locate relevant the relevant software help.

The implication of our findings is that children who wish to learn 3D design tools on their own, or who wish to continue learning and using these tools after a formal introduction to them in the classroom, face a formidable set of challenges. Thus, there is an unmet need for novel help and learning techniques to support children learning 3D design tools in individual settings. Specifically, reactive tutorials that can detect the success of individual actions and provide feedback would be valuable, as well as tutorial formats that better indicate relevant user interface elements.

Our main contribution is in providing empirical insights into the breakdowns that children face when learning 3D design software using currently-available instructional resources. Although we focus on 3D design software, we believe that our results can broadly apply to feature-rich software in other creative domains, such as graphic design and web design.

RELATED WORK
To contextualize our findings, we draw upon research on children's use of feature-rich software for creative and problem-solving tasks, including 3D design; children's help-seeking behaviors in other contexts, such as formal education; and children's online information-seeking behaviors.

Children's use of creative and problem-solving software
There is a long history of research in the area of child-computer interaction on designing and evaluating tools to support creative and problem-solving tasks [39], in domains such as programming [25, 28, 47], storytelling [40, 73], and gaming [13, 60, 69]. These works have identified several common usability issues faced by children (e.g., maintaining focus on a relevant part of an application, wandering into obscure parts of an application, and difficulty processing text-based instructions). We build on this work, with a focus on how commonly-available tutorials, and online help and learning resources, support children in learning 3D design software.

Although studying children's use of 3D design software is a relatively new focus in HCI research, recent work has explored "maker" activities with 3D printers in the classroom

and formal instruction settings [8, 22, 27], in special education and accessibility contexts [12, 42], and in ICT4D environments [70]. In addition to highlighting how 3D printing can be a valuable means of creative self-expression, these works have identified a range of usability challenges that children face with 3D design software and 3D printing hardware. However, this body of past work has generally been undertaken in settings where in-person assistance was available, and thus has had little to say about how children are supported in learning 3D design tools individually. Our study sheds light on children's learning of 3D design tools in individual settings, and provides specific design recommendations suitable for guiding the development of novel help and learning systems for 3D design, and creative and problem-solving tools more broadly.

Children's help-seeking behaviors in education
Although literature on how children learn and seek help when using new software is scant in HCI, several researchers have explored children's help-seeking behaviors in the education field. Different models have been proposed to better understand help-seeking among children, such as the one proposed by Nelson-Le Gall [33] for classroom settings, and later adapted by Aleven et al. [3] for electronic learning environments. These models provide a useful conceptualization of understanding the different stages of seeking help that include the following broad stages: *Become aware of need for help*; *Decide to seek help*; *Identify potential helper/resource*; *Use strategies to elicit help*; and *Evaluate the help-seeking episode*. Our findings shed light on these stages of help-seeking in the context of learning a new 3D design application in an individual setting.

Other key results from the education field suggest that children's help-seeking behaviors are affected by various motivational factors and attitudes [58, 67], peer support [59], and the involvement of instructors [67]. Recent works have also looked at integrated learning environments and how children seek help when using tools such as cognitive tutors [2, 48]. In both types of learning environments, children can struggle in monitoring task progress and assessing need for help [32, 57], along with other metacognitive challenges. In our discussion, we reflect on our key results about children learning software for recreational and creative use in individual settings, in relation to these prior findings from more formal scholastic learning environments.

Children's online information-seeking behaviors
Although not specific to the context of learning or troubleshooting software, there have been several studies on children's use of general online search and information-seeking strategies that have relevance to children's use of online help resources. For example, some studies shed light on the topics that children are interested in searching on the web and how the difficulty of searching varies by topic [19, 21, 62]. Others have explicitly focused on children's search and querying behaviors with online search engines [19, 20]. They have identified common problems experienced by children,

such as spelling mistakes in queries, challenges with formulating relevant queries, understanding returned results, and accessing non-relevant content. Even though custom search engines for children have been developed (e.g., Yahooligans), they appear to present similar challenges for children in finding and assessing relevant content [7]. Our findings complement these existing studies by investigating use of web search for software troubleshooting and learning tasks. We further discuss how children could be taught or encouraged to make better use of web-based resources and the rich technical help content that these resources contain.

In summary, our study complements the above body of literature with an evaluation of how commonly-available help-seeking behaviors and learning resources for 3D design software support children's use of these applications.

RESEARCH APPROACH

Our approach to this research is motivated by two main trends. First, there is a general lack of formal opportunities for children to learn and practice 3D design skills. Although some elementary and secondary schools are beginning to offer formal 3D design training [8, 37, 56], this practice is not widespread. Less formal methods, such as 3D design workshops at summer camps, makerspaces, and libraries [24] offer opportunities for children to learn the basics of this domain. However, since these workshops are relatively short, they may not offer the opportunity for longer-term learning and skill development. Second, a growing number of 3D design tools, such as *Tinkercad* [4], are becoming available for free on the web, as is an increasing body of online tutorials and help resources for these applications. In principle, this availability of tools and learning materials could provide children with all that they need to independently learn and develop skills in 3D design. However, the success of this strategy depends on whether the available resources can meet children's learning and help-seeking needs. Answering this question is the focus of this paper.

INTERVIEWS WITH WORKSHOP FACILITATORS

To inform the design of our one-on-one observational study, we conducted interviews with facilitators of 3D design workshops. Our goal was to draw on their experiences to understand what parts of learning 3D design children most often struggle with; how children seek help while learning; and which tools, tasks, and concepts children are being taught.

Method

We conducted semi-structured interviews with eight facilitators who had experience working with children in workshops focused on 3D design or 3D printing. Each interview lasted between 45 and 60 minutes. Interviews were audio recorded, transcribed, and then analyzed for common themes. Three of the facilitators also invited us for site visits, giving us an opportunity to observe children learning 3D design, and gain additional context for our semi-structured interview data.

We asked facilitators to characterize their workshops, including the typical number of participants; participants' age range, backgrounds, and motivations for taking part in the
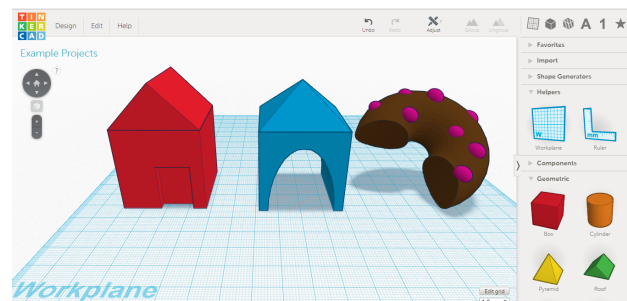

Figure 1. Screenshot of the Tinkercad user interface, showing several models used by workshop instructors

workshops; the format for instruction; and the software tools they taught. We also asked the facilitators to describe the main challenges they had observed that participants faced in using 3D design tools in the workshop setting. Finally, we asked the facilitators to describe any help-seeking behaviors they had observed participants engaging in during the workshops, and whether they had observed participants using common online help and learning resources, such as web search or in-application help.

We used an inductive analysis approach [15] to analyze the interview transcripts. Open coding was used to label transcript data, and affinity mapping was used to identify themes related to the format of the workshops, common breakdowns children experienced when learning 3D design software, and help-seeking behaviors in the workshop setting.

Results

Workshop and Participant Characteristics

The workshops described by the facilitators typically consisted of between 5 and 20 participants, and were targeted at children between 8 and 15 years old. Participants came from a range of cultural and economic backgrounds, representative of the neighborhoods in which the workshops were held. Participants typically did not have experience with 3D design or other "maker" activities prior to the workshops. Likewise, facilitators reported that the typical parent of a participant had little familiarity with 3D design, as in the quote below:

Most parents that bring their kids to the [workshop], most of them are like, 'I'm not handy at all, I can't make anything.' I'd say in a class of eight kids, there's probably one or two keen parents who know stuff. The rest are just like, 'My kid should be doing this because it's cool cutting-edge technology.' (F5)

This suggests that many children who take part in workshops will not be able to draw on their parents for continued support with learning at home once the workshop is finished.

The typical workshop described by the facilitators began with a live "follow-along" tutorial. The facilitator would project their screen and demonstrate how to construct an object step-by-step in the 3D design tool, while participants followed along at individual computers. The object used as a demonstration varied between facilitators (some examples are shown in Figure 1), but was chosen to include a representative set of 3D design operations, including aligning objects, resizing, and additive/subtractive geometry

operations. After the follow-along tutorial, students were typically allowed to pursue open-ended design exercises with help from the facilitator or an assistant as needed.

In terms of the 3D design tools taught in the workshops, Tinkercad was the choice of all facilitators. Tinkercad is a free, web-based 3D design tool that is popular among the maker community. It presents a simple interface (Figure 1), and supports modeling through a paradigm in which solid shapes are dragged into a work space, positioned and manipulated through direct manipulation, and combined with other objects through Boolean addition and subtraction methods. The simplicity of Tinkercad's interface was cited as the primary reason for choosing it over alternative tools.

Challenges in Introducing 3D Design Tools to Children
Facilitators described a range of common challenges faced by children when using 3D design software for the first time. Many of these difficulties had to do with working in a 3D design environment. For example, six of the eight facilitators mentioned that participants had difficulty with 3D camera control, and five of the eight facilitators mentioned that participants had difficulties conceptualizing 3D space:

> *Camera rotation is a really hard one for them. I feel like orientation of shapes is really hard too. [...] For example, they're looking at a shape straight on and they think they have added a shape on top [of it], but when you pan the camera, that shape is several inches behind it. (F4)*

Potentially related to this theme, most of the facilitators (7/8) mentioned that participants had trouble understanding what contributes to whether a model can be 3D printed or not.

Finally, six of the eight facilitators mentioned that participants experienced challenges with understanding how particular tools in the 3D design software operated.

Overall, the challenges discussed above are consistent with previously identified usability and learnability issues for 3D design software [41, 52].

Help-seeking Behavior in Workshops
The most common form of help-seeking that we observed during our site visits was participants asking their peers for help, or looking to other participants for inspiration. We also observed participants actively seeking out peers to provide help to, or enthusiastically sharing new features of the software that they had discovered. One facilitator summarized the low inhibition of participants in seeking or offering help:

> *They will ask other people, especially if they see somebody doing something successfully that they can't seem to do…in general, kids are awesome about helping each other. They'll totally stop their own project and go help the other kid. (F1)*

While these social help practices seemed positive in general, we also found some potential drawbacks. For example, three of the facilitators noted that children would sometimes hurt each other's feelings during social exchanges, and were not always effective teachers when they did try to provide help. As well, children who were shy were at a disadvantage, as they would sometimes run into trouble and not ask for help.

Participants also relied on the facilitators for help, with younger children relying more heavily on help from the facilitators while older children engaged in more peer-to-peer help. The facilitators we interviewed attributed this difference to older children having a better awareness of their own needs and the relevant skills or knowledge possessed by their peers.

Finally, we asked the facilitators about other potential help-seeking strategies that the children might use, including web search and other online help resources. The overwhelming response was that these were rare practices among workshop participants. One facilitator told us *"if a [child] is using the Internet in one of our workshops it's because they've gotten sidetracked"* (F1). Likewise, none of the facilitators had observed participants using Tinkercad's in-application help.

Summary
Overall, our interviews with facilitators illustrated a set of challenges and help-seeking behaviors used by children, and suggested that group workshops are an effective environment for introducing children to 3D design. However, not all children have access to these types of workshops, and even for those that do, there is a need for continued support once a workshop has ended. Moreover, we found that many parents of children who take part in the workshops were not themselves knowledgeable about 3D design tools, which suggests that children may not receive any ongoing support at home in learning 3D design. The above points suggest that it is important to understand the challenges faced by children learning 3D design software in individual settings.

ONE-ON-ONE OBSERVATIONAL LAB STUDY
Based on the insights gained from our interviews, we designed our observational study to investigate (1) how individual learning resources, such as web-based tutorials, video tutorials, and in-application tutorials support children's needs; and (2) whether, and how, common online help-seeking behaviors such as web search and online community help support children's needs in this domain.

Method
We conducted an observational study in which children individually followed tutorials to perform 3D design tasks. The overall approach was guided by prior work describing best practices for conducting user studies with children [1, 11, 18]. The study was approved by the Office of Research Ethics at the university at which the research was conducted.

We tested three tutorial types that are commonly used for software learning—static web pages with text and images, video tutorials, and in-application guided tutorials. Our intent was not to directly compare these three types of tutorials, but to gain insights on the appropriateness and trade-offs of each, specifically when used by children.

To investigate the effectiveness of various help-seeking behaviors, we employed a modified question-asking protocol [35]. In the standard question-asking protocol, a participant works on a task with an expert observing them. The participant is encouraged to ask the expert for assistance throughout

the process, which provides insights into the challenges faced by the participant, and their mental model of the system being studied. In our variation on this protocol, the experimenter directed the participant to different online help resources, to gain additional insights into the effectiveness of these resources and how they are used. When a participant asked a question, or was visibly stuck (indicated by repeating the same few incorrect actions; staring at the screen with no actions; expressing they were stuck more than once; or becoming visibly frustrated), the experimenter would prompt them to try to find the solution using either Google search or the Tinkercad knowledgebase (a search portal for Tinkercad's community forums). If the participant was still unsuccessful, the experimenter would provide answers and suggest actions to help the participant proceed with the study task.

In our pilot studies, we found that participants were often reluctant to ask the experimenter for help, so we adopted an approach in which the experimenter proactively observed the participant for signs that they were stuck, and prompted them to seek help. We also intentionally did not prompt participants to seek online help if they were visibly frustrated or had already failed to find help using a given resource several times. In these cases, we simply assisted the participant ourselves.

Participants

We recruited 15 participants (10 male, 5 female) in the 10–15 year age range. None of our participants had significant prior experience with 3D design software. Our rationale for selecting this particular age range was that children were likely to have reached Piaget's formal operational stage [6], and thus possess the deductive reasoning skills to make online help-seeking possible. We recruited from three main pools: a local children's sports team, children of members of a university mailing list, and children of employees of a large software company. Our entry interviews indicated that 6/15 participants had prior experience with 2D graphics software; 8/15 with Minecraft; 2/15 with web search (to find help with software); and 1/15 with web search (to find tutorials). Each study session took 45–60 minutes in total, and participants were given their choice of a small toy or a $25 gift card for participating.

Apparatus

The study sessions were held in a closed office where participants used a provided laptop computer (*15" Apple Macbook Pro*) with a USB mouse. Tinkercad was loaded in a Safari browser window and a separate browser tab contained the tutorial content. Participants were told that they were free to use the web to look up other content during the study session.

Study Tasks and Procedure

Each session began with a brief entry interview to gauge the participant's previous experience with computers, 3D design software, and web search. To build rapport, the experimenter also asked about the participant's hobbies, what they like and don't like at school, etc.—a measure that we found was effective at putting the participant at ease. Next, the participant was asked to perform a series of short tutorials built into the Tinkercad application, to familiarize themselves with the basics of camera movement, moving objects, and subtractive
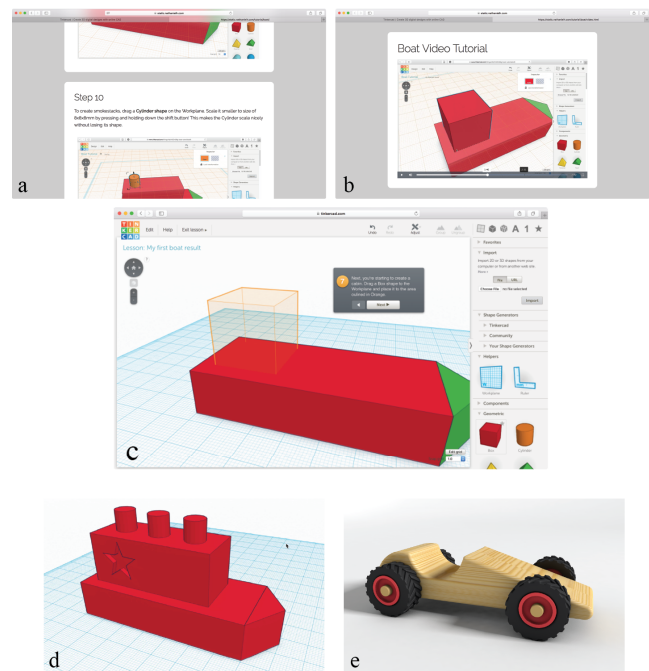
Figure 2. (a) text-and-images tutorial, (b) video tutorial, (c) in-application tutorial, (d) boat model, (e) toy car model.

geometry. These tutorials launch automatically when the software is used for the first time, so this emulates a "first contact" experience with the software. Next, each participant performed a tutorial that walked them through the process of 3D modeling a boat (Figure 2d). To capture insights into the full range of tutorial types typically found on the web, we employed three versions of this tutorial—a static web page with text and images (Figure 2a); a video tutorial with a screen recording and voiceover (Figure 2b); and an in-application interactive tutorial that provided text instructions augmented with outline indicators displayed in the workspace for some steps (Figure 2c). The in-application tutorial was a preexisting Tinkercad feature, whereas the other tutorials were developed specifically for this study by translating the existing tutorial into these other formats. Instructional text and instructions were kept as consistent as possible across the three tutorial types, though modifications were made to adapt the instructions to each format. We balanced the assignment of tutorial types to participants (five participants to each type), and were careful to balance the ages of participants assigned to each of the three groups. All fifteen participants successfully completed the boat tutorial, with varying levels of assistance from the experimenter, suggesting that the task was at a reasonable level of difficulty. If time permitted, participants could work on an additional, more challenging task—we gave them an image of a toy car (Figure 2e), and asked them to re-create it. Ten participants attempted this task, producing models of varying quality.

Video and screen recordings were collected for each session. We also manually recorded the types of help-seeking questions asked by participants.

Data Analysis

To analyze the gathered data, we used an inductive analysis approach, drawing on methods from grounded theory [15]. Screen and audio recordings were reviewed by the primary experimenter to identify (1) instances where participants faced challenges (with the software, tutorials, or help-seeking approaches), (2) instances of problem-solving strategies employed to overcome challenges, and (3) instances where particular help-seeking approaches were used. The resulting observations were labelled using an open coding approach, and clustered using affinity mapping to identify common themes for each of the above areas. This process was led by the primary experimenter, with ongoing input from the other researchers in interpretation and synthesis sessions. Input from the other researchers included independent examinations of the data to help refine the coding scheme, and discussions of emerging themes, which led to additional ideas on how to cluster the data, or ideas on new themes that may explain the observed data. Through the above process, all of the researchers collaboratively developed the resulting coding scheme and clustering, while checking its legitimacy throughout the process. Over several iterations, the themes presented in the following sections emerged.

Challenges Following Instructional Tutorials

We start by presenting a comparison of the challenges we observed in participants using the three tutorial types, as well as broader themes we observed in how participants approached the tutorials. Overall, we did not observe any one of the tutorial types to be clearly superior to the others, though there were differences in how each was used by the participants. Table 1 summarizes the most common challenges we observed for the three tutorials.

Challenges in locating referenced UI elements

Participants struggled to locate user interface elements referenced in tutorials, such as buttons or panels, an issue that has been previously documented in the context of learning feature-rich software [35]. This was particularly a problem for Tinkercad's in-application tutorials, which at the time of this study did not provide screenshots or other visuals that could aid in locating referenced user interface elements.

Overlooking instructions, or performing them incorrectly

Several of the challenges that we observed relate to how participants consumed tutorial steps, and how this led them to overlook instructions, or perform instructions incorrectly.

In the video tutorial, there was no explicit demarcation of individual steps, but there were pauses after individual actions, occurring every 10–15 seconds. Three of the five video participants watched several of these "steps" before switching back to the application and attempting to perform all of them together. This frequently led to these participants omitting required actions, or performing actions incorrectly.

This suggests that these participants overestimated their ability to recall and carry out the actions they had observed. It may also suggest that the explicit grouping of actions into steps (in the non-video tutorials) provides natural break points that

	Text/Image	Video	In-app.	Total
Overlooking instructions or details of a step	4/5	2/5	2/5	8/15
Locating referenced user interface elements	2/5	1/5	4/5	7/15
Not moving the camera	1/5	2/5	4/5	7/15
Attempting to perform multiple steps together	0/5	3/5	0/5	3/15
Ignoring written instructions altogether	1/5	N/A	2/5	3/10

Table 1. Common challenges observed for the three tutorial types, and the number of participants that exhibited each.

prompt the participant to return to the application and carry out a set of actions. However, formats that included explicit steps were challenging for participants as well, as we discuss next.

In the text-and-image tutorials, 7/15 steps described multiple actions, and participants frequently overlooked some of the included actions. For example, one tutorial step read: *"To give some unique touch for the boat, let's drag a Star shape on the Workplane. Push the Star a little bit into the cabin. Choose the Hole feature in the up right corner of the editor."* Six of the fifteen participants performed the first action (dragging the Star shape onto the workplane) then moved on, overlooking the remaining two actions. In other cases, participants missed helpful tips at the end of a step. For the step: *"To create smokestacks, drag a Cylinder shape on the Workplane. Scale it smaller to size of 8×8×8mm by pressing and holding down the shift button! This makes the Cylinder scale nicely without losing its shape"*, nine of fifteen participants missed the faster method of holding shift documented at the end of the step, and manually scaled all three axes of the cylinder individually.

Finally, for the text-and-image and in-application tutorials, some participants stopped attending to the written instructions altogether, and instead focused on the images (in the case of the text-and-image tutorials), or in-workspace guides (in the case of the in-application tutorials). In the case of the in-application tutorials this was a particular source of difficulty. The in-workspace guides—semi-transparent outlines in the workspace that indicate where to position elements—frequently led participants to position objects so they looked correct from the current camera perspective, but were in fact not lined up correctly in 3D space.

One explanation for the above observations is that the children read the first part of the step, or observed the images or in-workspace guides, and assumed that they understood what they needed to do to perform that step, causing them to start acting without considering the instructions more carefully. This could also be based on a presumption that individual tutorial steps will represent one action each.

Participant comments also suggested a high initial confidence with tutorials, but later disappointment, as explained by one participant: *"[The tutorial] should have had somebody new doing it, because then it would show how it could actually be hard. They made it look easy, but it's disappointing when you don't get it right." (P13)*

Imitating actions in tutorials

Another phenomenon we observed was that the children would mimic actions they had observed in tutorials, even without a full understanding of why those actions were necessary. This was apparent when participants imitated actions from tutorials in contexts where these actions were not required, suggesting the participant had formed an incorrect or incomplete understanding of the tool. While imitation often resulted in unnecessary actions, we did observe cases where this imitation behavior was beneficial and promoted best practices, as we describe below.

We observed that children who had seen camera movement in video or text-and-image tutorials would more frequently move the camera to look at the model from different views, imitating the behavior they had seen in the tutorials. This imitation behavior helped the participant observe problems with alignment in the model they were creating. The tutorials contained no explicit instruction to move the camera, and the discovery of alignment issues often came as a surprise to the participants, further suggesting that they were not using the camera movement to explicitly check that shapes were aligned properly, but were instead simply imitating camera adjustments observed in the tutorials. This behavior is important, because not moving the camera to look at a model from different angles is a well-documented source of difficulty for novice users of 3D design tools [41]. Moreover, we observed this challenge in our study as well, with four of the five children in the in-application tutorial condition experiencing challenges stemming from not moving the camera to view the scene from multiple angles.

While imitation of camera movement tended to yield positive results, we observed other cases where imitation behavior led to difficulties, such as with use of the workplane tool. The boat tutorial performed by all participants included several steps that required moving the workplane, but when asked, eight of eleven[1] children reported not understanding the purpose of the workplane tool. Despite this lack of understanding, five of the ten participants who performed the car task moved the workplane in situations where there was no advantage or clear reason for doing so. These participants reported that they were attempting to mimic best practices, and expressed that "*the tutorial said that's the right way to do things*" (P8). These imitation attempts would often create new difficulties for these participants, as it would cause object transformations to happen in unexpected ways.

In summary, participants struggled with all the types of tutorials we tested, with many of their challenges stemming from overlooking actions in tutorial steps or choosing to attend to only parts of the full guidance provided. Participants also mimicked actions or perceived best practices they had observed in tutorials, which was sometimes beneficial but led to difficulties as well.

---

[1]Initially, we did not realize that the workplane tool would be a point of difficulty for participants, so not all participants were asked about it.

Question Type	Count
Procedural Help (e.g., *How do I do this…*)	67 (57%)
Determine Possibilities (e.g., *Can I do this…*)	29 (25%)
Locate in Interface (e.g., *Where is this…*)	17 (14%)
Terminology (e.g., *What does this mean…*)	5 (4%)

Table 2. Most common types of help requested.

Individual Help-seeking Behaviors

We next present our observations of participants' use of online help-seeking methods, the challenges that participants faced in locating and evaluating help resources, and a general negativity bias that we observed toward online help-seeking by the participants in the study. Before we present these findings, we start by summarizing the triggers for help-seeking episodes, which provide context for the subsequent findings.

Types of Questions Asked by Participants

As summarized in Table 2, the most common questions that we received from participants were requests for procedural help (57%). In these instances, the participant was aware of what they needed to do, and that the software had the capability to accomplish that goal, but they were not clear on the specific actions they needed to take. For example, one participant read the instruction "*Drag the workplane on top of the Boat's body*", then asked for help with how to accomplish this in the software.

In other cases, participants' questions were about awareness of functionality [35], including whether a particular task was possible or whether particular features existed (25%). For example, one participant asked if there was a tool in Tinkercad to "round over" shapes (typically known as a "fillet" in 3D software, a capability that Tinkercad did not have at the time).

Other common types of questions concerned locating specific user interface elements in the application (14%), or requests for help with understanding terminology (5%).

Challenges with Online Help-seeking

Although adults can often benefit from making coordinated use of online resources when working with feature-rich software [23, 30], none of the children in our study used web search or forums without being prompted. In our entry interviews, only two of the children reported having used web search in response to software difficulties in the past, and only one reported having used a forum to troubleshoot software.

When we received a request for help from a participant we directed them to two help resources—Google Search, and the Tinkercad Knowledgebase—alternating between the two on a per-request basis. Web searches led participants to a variety of resources, including text-and-image tutorials (32%), blog posts (25%), video tutorials (18%), forums (14%), and other resources (11%). In general, we observed that none of these resources were broadly effective at helping participants to accomplish their goals. If we define success as cases where the participant found the information they needed and applied it successfully in the application, Google Search was successful in 17% of cases, and the Tinkercad Knowledgebase was successful in 5% of cases (Figure 3).
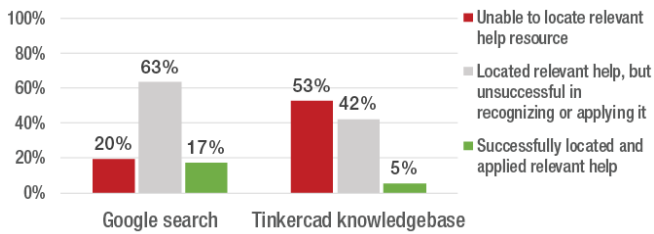
Figure 3. Summary of outcomes for help-seeking instances.

We observed two main ways that help-seeking failed – the participant failed to locate content that contained a solution for the challenge they were facing; or participants located a solution, but failed to recognize its relevance or how it could be put into action. To quantify this behavior, we recorded the webpages participants visited and had a member of the research team who was familiar with Tinkercad evaluate whether each page contained a relevant solution. We observed that even though participants' search queries varied widely in quality (discussed below), Google returned relevant results 80% of the time (Figure 3). However, participants only recognized and successfully applied the information they found in 17% of help-seeking instances. Participants were even less successful when using the Tinkercad Knowledgebase, only locating potentially helpful resources in 42% of instances, and successfully using this information in 5% of instances.

Below we provide a more detailed discussion of the challenges that the children faced in formulating effective search queries for technical help and evaluating returned resources.

Challenges in Locating Relevant Help Resources
Over the study, participants issued 53 search queries in total (an average of 4 per participant). Consistent to what prior studies have found about how children search online in other contexts [19], the participants in our study tended to search using natural language questions such as *"Does command z work on tinker cat"* (sic) for technical help queries as well.

Participants experienced difficulties with terminology and spelling while formulating queries, which is consistent with prior work [19]. However, we were surprised by how Google's autocorrect feature frequently caused terminology problems for participants as they tried to describe application-specific commands and features. For example, one participant who had previously correctly used the term "workplane" in conversation, issued the search query *"tinkercad move the workplne"*, which google corrected to *"tinkercad move the workplace"*. The participant accepted the suggested incorrect terminology without question, and then began to use the incorrect term when speaking with the experimenter. This suggests that this child viewed search results, or this spelling correction feature more specifically, as trustworthy or an authority, and that extra caution is needed around the use of generalized autocorrect for children when they work with domain-specific terminology.

Search-query suggestions during query formulation were also a source of distraction for children as they tried to complete the 3D design tasks. For example, autosuggest results such as,

"how to be single" and *"how to play Pokemon Go"* prompted fits of laughter from the children, and the experimenter often had to discourage them from following these suggestions. This aligns with what we heard from the workshop facilitators, who suggested that online resources were more a "source of distraction" than of help for children in their 3D design workshops. (In contrast, this distraction was less of an issue on the application-specific knowledgebase, likely due to the more focused nature of the content it contained).

Finally, as seen in other studies with children and keyword search [19, 20], our participants also had difficulty formulating abstract help queries for technical help. In several cases participants issued queries that were overly specific to their problem with an application feature or functionality (e.g., *"how to resize a cylinder to 6mm by 6mm by 6mm in tinkercad"*). In many cases the wording of these specific queries was copied directly from the tutorial. The specificity of these queries limited the quality of the returned results. This was particularly problematic when using the less sophisticated search system provided by the knowledgebase.

Challenges in Evaluating Help
In addition to challenges with query formulation, we observed that children had difficulty evaluating the relevance of help resources, a problem that has been identified in other contexts for children [38]. Although in many cases the children located web pages that contained a relevant answer, they often failed to recognize that a page did contain a solution to their application-specific problem.

False rejection was particularly problematic when using the knowledgebase, as deciphering the back-and-forth forum discussions would often leave participants confused or irritated. One participant expressed frustration with the forum posters, saying *"Why would I want to use a forum? […] It's just people arguing with each other." (P5)* Other children refused to use the community help resources altogether, as their parents had warned them not to trust what strangers say online, as in the following comment:

> *Oh, [the knowledgebase] is just people. I don't want this. […] My dad said not to trust people [online] unless they've got [credentials] that says that they're smart. (P15)*

In many cases, false rejection occurred when search results returned introductory tutorials (both official tutorials associated with Tinkercad, and community-created). The children also had difficulty skimming longer pages to find information, and would often reject such pages as being unhelpful.

Conversely, children would often spend considerable time evaluating search results that contained no useful information, but contained potentially-relevant keywords and images. For example, API documentation, very old documentation, and documentation for other software would often contain keywords relevant to Tinkercad and images that would lead participants to believe that they may be helpful. One participant, while trying to determine how to move objects vertically, spent several minutes trying to decipher the "Tinkercad Shape Generator Overview", which is a developer

API allowing for new shapes to be programmed using a specialized scripting language. She pointed to an image containing arrows overlaid on a cube, and the word "overview", as evidence that the page must contain information about how to move objects.

Negativity Bias Toward Help-seeking Methods

Due to their lack of success, participants quickly formed negative opinions of the help-seeking methods we prescribed as part of our study method. In several cases, participants made exclamations such as "*Oh dammit, I don't want to do this again!*" *(P8)* when directed to these resources a second time. The participant who made this comment went on to ignore the experimenter's suggestion to seek help online, and continued to try and find a solution in the software's interface instead. This may suggest that a bad experience caused by one of the factors discussed above can cause children to quickly form a negative opinion of a given help-seeking method, or decide to reject its usefulness entirely.

DISCUSSION

Our key results (summarized in Figure 4) indicate that children struggle with effectively utilizing common online learning and help-seeking resources, such as tutorials, web search, and community forums, which could potentially support them in independently learning 3D design software.

Below, we contextualize our key findings in terms of existing theoretical and empirical results on children's help-seeking behaviors, and discuss our study's new insights into how children seek help and learn software for recreational and creative use. We discuss several directions for future work to explore the design of next-generation help and learning systems for children using 3D design software and beyond.

The Larger Context of Children's Help-Seeking Behaviors

One of the key challenges that children faced with tutorials was that they tended to overestimate their understanding of the intent of individual tutorial steps, causing them to overlook required actions, or in the case of video tutorials, watch a large amount of tutorial content and then attempt to switch back to the application to try and carry it all out. They rarely asked for help in such situations, even when they were visibly stuck. This could be explained by results from past research that suggest that a learner must engage in higher-order metacognitive functions in recognizing the difficulty of completing the task and monitoring task progress before the need for help can be assessed [32, 57]. As children's meta-cognitive abilities are still in development, they can struggle in monitoring and reflecting on their performance, and their need to seek help [10].

When children were prompted to search for help, some of their struggles were similar to issues known to be faced by adults trying to describe software problems (e.g., formulating queries using appropriate vocabulary [31]). However, such issues appeared to be more acute for children. For example, we found that search features such as query suggest and autocorrect could mislead children or become a source of distraction, corroborating previous findings on children's struggles with information-seeking in other contexts [19, 20]. In instances

Challenges following tutorial instructions
- *Overlooking or ignoring actions in tutorial steps*
- *Locating referenced UI elements*

Help-seeking challenges
- *Reluctance to initiate help-seeking*
- *Difficulty formulating effective search queries* (spelling/terminology issues, difficulty expressing abstract needs)
- *Evaluating whether a help-resource will meet their needs* (particularly for long and complex resources, such as forum posts)

Figure 4. Summary of the key challenges faced by children learning 3D design, based on our study observations.

where children did locate a relevant help resource, we found that they faced challenges with evaluating the usefulness of the resource, and with putting the found knowledge into practice.

Another key observation of children's help-seeking behaviors was their quick formation of negative opinions of help resources, or tendency to reject them entirely in response to an unsuccessful help-seeking experience. This behavior is consistent with other studies of help-seeking where children ignore or do not make effective use of on-demand help in classrooms [67] and integrated learning environments [3]. This behavior is also consistent with more theoretical explanations on how children may avoid seeking help due to the additional cognitive load it imposes (i.e., they find it difficult to simultaneously work on an unfamiliar task and engage in help-seeking [71]).

Given the help-seeking challenges our study uncovered, future work should investigate the extent to which help-seeking evolves with age, particularly in recreational and individual use of feature-rich software.

Design Implications

Several research projects have been undertaken to make 3D design more accessible to children, through novel interfaces for 3D modeling, or new interactions with 3D environments (e.g., [29, 43, 51]). Our study complements this work, identifying additional opportunities to support children learning 3D design software by rethinking the design of help and learning systems. Furthermore, many of our results are likely generalizable to children's interactions with software beyond 3D design. We discuss several of these opportunities below.

Reactive and Integrated Tutorials

Given that children's challenges with help and learning resources may be related to higher-order metacognitive functions, it would be valuable to investigate systems that help develop these metacognitive abilities, or that stand-in for them while they are developing. For example, an integrated tutorial system could provide feedback on steps the user has completed in a tutorial, helping the user to reflect on their progress. This approach could address the tendency for the children to attempt to perform multiple tutorial steps together, to skip steps, and to overlook instructions within a given step. Along similar lines, reactive tutorial systems that enforce pacing, such as the Pause-and-Play video tutorial system [61], could be beneficial, as could tutorials that only reveal the next step once the previous step has been completed.

More broadly, tutorial systems could be developed that model the learner's knowledge, and generate or adapt tutorial content accordingly—an approach that has been investigated in the programming domain [36]. For 3D design, this could operate by identifying specific skill deficits (e.g., in 3D navigation or Boolean operations) and providing tasks designed to practice or improve these skills.

One potentially valuable application of this approach would be to encourage best practices. We observed that children would often imitate practices they viewed in tutorials, even if they did not always understand the intent of the practice. Moreover, not following best-practices was a source of difficulty for participants. Adaptive tutorial systems [26] could be designed to recognize when a user is not following best practices (e.g., an extended period of not moving the camera in 3D design software, or not checking for compile errors in programming tasks), and dynamically insert steps that prompt the learner to perform these actions. Likewise, the tutorial system could hide or de-emphasize such instructions once a learner has demonstrated that they have acquired the relevant skill.

A similar approach could be applied to the challenge of unfamiliar vocabulary and terminology, which we observed in our study, and which is likely occur in many other complex software domains as well. The user's knowledge of domain-specific terminology could be modelled by the system, and tutorial content could be dynamically adapted to show or hide supplementary help and explanations of key terms.

Social Help-seeking for Individual Learning Settings
Given the enthusiasm we observed among children engaging in social help-seeking and learning during group workshops, it would be interesting to try and recreate these benefits for children learning in individual settings, potentially by connecting individual learners with a broader community of users. However, our one-on-one study indicated that children had difficulty establishing trust in online help resources or accessing information from forums. Thus, any community-based help system for children should be developed with mechanisms for establishing trust as a primary design goal.

The *Scratch* community [44, 63] has had impressive success with connecting children with each other when learning programming, and best practices from that project may be applicable to the 3D design domain as well.

Improving Search Systems for Finding Help
Natural language search has proven to be challenging for children in a range of contexts [19, 20, 34, 38], and our findings confirm that it is a challenge for children seeking software help as well. To address these challenges, a range of search systems have been developed to help children formulate queries (e.g., visual query formulation systems [45, 65]), which may be able to be applied to help children formulate queries when seeking help with 3D design or other visual software. Automatic query expansion [14, 16] is another approach that could be applied to this challenge.

Beyond formulating queries, a key challenge faced by the children in our study was recognizing that they had found relevant

help, and successfully applying the found information. Thus, it may be valuable to explore systems that could assist children in evaluating relevant help and learning materials. For example, a system that integrates the user's web browser with the 3D design application being used could help them evaluate the relevance of resources in the context of the document being worked on in the application. Similar ideas have been explored in past work [23, 30], but we are unaware of this idea being applied to children or the 3D design software domain.

One caution with the above is that the children who participated in our study did not use search tools to seek help unless prompted, and typically had little or no prior experience using the web as a source of technical help. Unlike adults [5, 9, 66], children in our study did not perceive search as a fruitful medium for seeking troubleshooting or learning help. This may suggest a need for systems that explicitly introduce and teach search skills, or suggest search as a strategy to the user (e.g., by detecting that the user is struggling). Another potential approach is to monitor use of the application, and automatically present help and learning resources to the user as they work. This idea has been explored in prior work targeted at professional users of 3D design software [55], but may be more effective for children who are used to receiving ambient help and instruction from teachers, parents, and caregivers.

Limitations
Our sampling methods may have introduced bias by over-sampling from specific communities, or children of parents who are interested in technology. It is also possible that we recruited participants with low intrinsic motivation to learn 3D design, and this impacted their help-seeking effectiveness (past work has provided some evidence for a connection between motivation and help-seeking effectiveness in children [17]). However, in our observations participants were eager to complete the study tasks, and we do not believe that motivation alone can explain all the challenges we observed.

Beyond issues of sampling, running studies with children is complex, and the presence of the researcher may have influenced their behavior, as may have the lab setting in which the studies were carried out, and the specific design of our study. Finally, while we made our best efforts to construct tutorials representative of those found online for 3D design tools, we only tested tutorials for one application.

CONCLUSIONS
As 3D design tools are increasingly being used by non-professional users, including children, hobbyists, and casual makers, there is a need to refine and update our understanding of how to design software help and learning tools to accommodate these new user populations. This paper provides a first step toward establishing a better understanding of children's learning and help-seeking behaviors with 3D design software, and we believe that our findings can serve as a foundation for developing next-generation help and learning systems to enable children to learn and improve 3D design skills.

ACKNOWLEDGEMENTS

REFERENCES

1. Houman Alborzi, Allison Druin, Jaime Montemayor, Michele Platner, Jessica Porteous, Lisa Sherman, Angela Boltman, Gustav Taxén, Jack Best, Joe Hammer, Alex Kruskal, Abby Lal, Thomas Plaisant Schwenn, Lauren Sumida, Rebecca Wagner, and Jim Hendler. 2000. Designing StoryRooms: Interactive Storytelling Spaces for Children. In *Proceedings of the 3rd Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques* (DIS '00), 95–104. https://doi.org/10.1145/347642.347673

2. Vincent Aleven, Bruce Mclaren, Ido Roll, and Kenneth Koedinger. 2006. Toward Meta-cognitive Tutoring: A Model of Help Seeking with a Cognitive Tutor. *Int. J. Artif. Intell. Ed.* 16, 2: 101–128.

3. Vincent Aleven, Elmar Stahl, Silke Schworm, Frank Fischer, and Raven Wallace. 2003. Help Seeking and Help Design in Interactive Learning Environments. *Review of Educational Research* 73, 3: 277–320. https://doi.org/10.3102/00346543073003277

4. Autodesk. Tinkercad | Create 3D digital designs with online CAD. *Tinkercad*. Retrieved July 28, 2017 from https://www.tinkercad.com/

5. Rob Barrett, Eser Kandogan, Paul P. Maglio, Eben M. Haber, Leila A. Takayama, and Madhu Prabaker. 2004. Field studies of computer system administrators: analysis of system management tools and practices. In *Proceedings of the 2004 ACM conference on Computer supported cooperative work*, 388–395.

6. Kathleen Stassen Berger. 2007. *The Developing Person Through the Life Span*. Worth Publishers, New York.

7. Dania Bilal. 2000. Children's use of the Yahooligans! Web search engine: I. Cognitive, physical, and affective behaviors on fact-based search tasks. *Journal of the American Society for Information Science* 51, 7: 646–665. https://doi.org/10.1002/(SICI)1097-4571(2000)51:7<646::AID-ASI7>3.0.CO;2-A

8. Paulo Blikstein. 2013. Digital fabrication and 'making' in education: The democratization of invention. *FabLabs: Of machines, makers and inventors*: 1–21.

9. Joel Brandt, Philip J. Guo, Joel Lewenstein, Mira Dontcheva, and Scott R. Klemmer. 2009. Two studies of opportunistic programming: interleaving web foraging, learning, and writing code. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1589–1598.

10. Ann L. Brown, John D. Bransford, Roberta A. Ferrara, and Joseph C. Campione. 1983. Learning, remembering, and understanding. In *Handbook of Child Psychology*, P. Mussen (ed.). Wiley.

11. Amy Bruckman, Alisa Bandlow, and Andrea Forte. 2002. *HCI for kids*.

12. Erin Buehler, Shaun K. Kane, and Amy Hurst. 2014. ABC and 3D: opportunities and obstacles to 3D printing in special education environments. In *Proceedings of the 16th international ACM SIGACCESS Conference on Computers & Accessibility*, 107–114.

13. Véronique Celis, Jelle Husson, Vero Vanden Abeele, Leen Loyez, Lieven Van den Audenaeren, Pol Ghesquière, Ann Goeleven, Jan Wouters, and Luc Geurts. 2013. Translating preschoolers' game experiences into design guidelines via a laddering study. In *Proceedings of the 12th International Conference on Interaction Design and Children*, 147–156.

14. Paul-Alexandru Chirita, Claudiu S. Firan, and Wolfgang Nejdl. 2007. Personalized query expansion for the web. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, 7–14. http://dl.acm.org/citation.cfm?id=1277746

15. Juliet Corbin and Anselm Strauss. 2007. *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*. Sage Publications, Inc.

16. Hang Cui, Ji-Rong Wen, Jian-Yun Nie, and Wei-Ying Ma. 2002. Probabilistic query expansion using query logs. In *Proceedings of the 11th international conference on World Wide Web*, 325–332. http://dl.acm.org/citation.cfm?id=511489

17. Dale Dougherty. 2013. The Maker Mindset. In *Design, make, play: Growing the next generation of STEM innovators*, Margaret Honey and David E. Kanter (eds.). Routledge.

18. Allison Druin (ed.). 1998. *The Design of Children's Technology*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

19. Allison Druin, Elizabeth Foss, Leshell Hatley, Evan Golub, Mona Leigh Guha, Jerry Fails, and Hilary Hutchinson. 2009. How children search the internet with keyword interfaces. In *Proceedings of the 8th International conference on interaction design and children*, 89–96. http://dl.acm.org/citation.cfm?id=1551804

20. Allison Druin, Elizabeth Foss, Hilary Hutchinson, Evan Golub, and Leshell Hatley. 2010. Children's roles using keyword search interfaces at home. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 413–422.

21. Sergio Duarte Torres and Ingmar Weber. 2011. What and how children search on the web. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, 393–402. http://dl.acm.org/citation.cfm?id=2063638

22. Michael Eisenberg. 2013. 3D printing for children: What to build next? *International Journal of Child-Computer Interaction* 1, 1: 7–13.

23. Michael Ekstrand, Wei Li, Tovi Grossman, Justin Matejka, and George Fitzmaurice. 2011. Searching for software learning resources using application context.

In *Proceedings of the 24th annual ACM symposium on User interface software and technology* (UIST '11), 195–204. https://doi.org/10.1145/2047196.2047220

24. Sandra Feinberg and James Robert Keller. 2010. *Designing space for children and teens in libraries and public places*. American Library Association.

25. Ylva Fernaeus and Jakob Tholander. 2006. Designing for programming as joint performances among groups of children. *Interacting with Computers* 18, 5: 1012–1031.

26. Jennifer Fernquist, Tovi Grossman, and George Fitzmaurice. 2011. Sketch-sketch revolution: an engaging tutorial system for guided sketching and application learning. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, 373–382. http://dl.acm.org/citation.cfm?id=2047245

27. Daniel Fitton, Janet C. Read, and John Dempsey. 2015. Exploring children's designs for maker technologies. In *Proceedings of the 14th International Conference on Interaction Design and Children*, 379–382. http://dl.acm.org/citation.cfm?id=2771921

28. Louise P. Flannery, Brian Silverman, Elizabeth R. Kazakoff, Marina Umaschi Bers, Paula Bontá, and Mitchel Resnick. 2013. Designing ScratchJr: support for early childhood learning through computer programming. In *Proceedings of the 12th International Conference on Interaction Design and Children*, 1–10.

29. Sean Follmer and Hiroshi Ishii. 2012. KidCAD: digitally remixing toys through tangible tools. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2401–2410. http://dl.acm.org/citation.cfm?id=2208403

30. Adam Fourney, Ben Lafreniere, Parmit K. Chilana, and Michael Terry. 2014. InterTwine: Creating interapplication information scent to support coordinated use of software. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology* (UIST '14), 429-438.

31. G. W. Furnas, T. K. Landauer, L. M. Gomez, and S. T. Dumais. 1987. The Vocabulary Problem in Human-system Communication. *Commun. ACM* 30, 11: 964–971. https://doi.org/10.1145/32206.32212

32. Sharon Nelson-Le Gall. 1981. Help-seeking: An understudied problem-solving skill in children. *Developmental Review* 1, 3: 224–246. https://doi.org/10.1016/0273-2297(81)90019-8

33. Sharon Nelson-Le Gall. 1985. Chapter 2: Help-Seeking Behavior in Learning. *Review of research in education* 12, 1: 55–90.

34. Tatiana Gossen, Juliane Höbel, and Andreas Nürnberger. 2014. A comparative study about children's and adults' perception of targeted web search engines. In *Proceedings of the SIGCHI*

Conference on Human Factors in Computing Systems, 1821–1824.

35. Tovi Grossman, George Fitzmaurice, and Ramtin Attar. 2009. A Survey of Software Learnability: Metrics, Methodologies and Guidelines. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '09), 649–658. https://doi.org/10.1145/1518701.1518803

36. Kyle J. Harms, Dennis Cosgrove, Shannon Gray, and Caitlin Kelleher. 2013. Automatically Generating Tutorials to Enable Middle School Children to Learn Programming Independently. In *Proceedings of the 12th International Conference on Interaction Design and Children* (IDC '13), 11–19. https://doi.org/10.1145/2485760.2485764

37. A. Hira, C. H. Joslyn, and M. M. Hynes. 2014. Classroom makerspaces: Identifying the opportunities and challenges. In *2014 IEEE Frontiers in Education Conference (FIE) Proceedings*, 1–5. https://doi.org/10.1109/FIE.2014.7044263

38. Sandra G. Hirsh. 1999. Children's relevance criteria and information seeking on electronic resources. *Journal of the Association for Information Science and Technology* 50, 14: 1265.

39. Juan Pablo Hourcade. 2015. *Child-Computer Interaction*. CreateSpace Independent Publishing Platform.

40. Juan Pablo Hourcade, Benjamin B. Bederson, and Allison Druin. 2004. Building KidPad: an application for children's collaborative storytelling. *Software: Practice and Experience*, 34(9), 895-914.

41. Nathaniel Hudson, Celena Alcock, and Parmit K. Chilana. 2016. Understanding Newcomers to 3D Printing: Motivations, Workflows, and Barriers of Casual Makers. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 384–396.

42. Amy Hurst and Shaun Kane. 2013. Making making accessible. In *Proceedings of the 12th International Conference on Interaction Design and Children*, 635–638.

43. Takeo Igarashi, Satoshi Matsuoka, and Hidehiko Tanaka. 1999. Teddy: A Sketching Interface for 3D Freeform Design. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques* (SIGGRAPH '99), 409–416. https://doi.org/10.1145/311535.311602

44. Yasmin Kafai, Ricarose Roque, Deborah Fields, and Andres Monroy-Hernandez. 2011. Collaboration by choice: Youth online creative collabs in Scratch. In *Proceedings of the 19th International Conference on Computers in Education*, 189–193.

45. Nancy Kaplan, Yoram Chisik, Kendra Knudtzon, Rahul Kulkarni, Stuart Moulthrop, Kathryn Summers, and Holly Weeks. 2004. Supporting sociable literacy in

the international children's digital library. In *Proceedings of the 2004 conference on Interaction design and children: building a community*, 89–96.

46. Caitlin Kelleher and Randy Pausch. 2005. Stencils-based tutorials: Design and evaluation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '05), 541–550. https://doi.org/10.1145/1054972.1055047

47. Caitlin Kelleher, Randy Pausch, and Sara Kiesler. 2007. Storytelling alice motivates middle school girls to learn computer programming. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1455–1464.

48. Kenneth R. Koedinger and Vincent Aleven. 2007. Exploring the assistance dilemma in experiments with cognitive tutors. *Educational Psychology Review* 19, 3: 239–264.

49. Benjamin Lafreniere, Andrea Bunt, and Michael Terry. 2014. Task-centric interfaces for feature-rich software. In *Proceedings of the 26th Australian Computer-Human Interaction Conference* (OZCHI '14), 49–58.

50. Benjamin Lafreniere, Tovi Grossman, and George Fitzmaurice. 2013. Community enhanced tutorials: Improving tutorials with multiple demonstrations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '13), 1779–1788. https://doi.org/10.1145/2470654.2466235

51. Ben Leduc-Mills and Michael Eisenberg. 2011. The UCube: a child-friendly device for introductory three-dimensional design. In *Proceedings of the 10th International Conference on Interaction Design and Children*, 72–80. http://dl.acm.org/citation.cfm?id=1999039

52. Ghang Lee, Charles M. Eastman, Tarang Taunk, and Chun-Heng Ho. 2010. Usability principles and best practices for the user interface design of complex 3D architectural design and engineering tools. *International Journal of Human-Computer Studies* 68, 1–2: 90–104. https://doi.org/10.1016/j.ijhcs.2009.10.001

53. Hod Lipson and Melba Kurman. 2013. *Fabricated: The new world of 3D printing*. John Wiley & Sons.

54. Justin Matejka, Tovi Grossman, and George Fitzmaurice. 2011. IP-QAT: In-product Questions, Answers, & Tips. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology* (UIST '11), 175–184. https://doi.org/10.1145/2047196.2047218

55. Justin Matejka, Tovi Grossman, and George Fitzmaurice. 2011. Ambient help. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '11), 2751–2760. https://doi.org/10.1145/1978942.1979349

56. Christian McKay and Kylie Peppler. 2013. MakerCart: a mobile fab lab for the classroom. *Position Paper at the Interaction Design for Children Conference (IDC)*.

57. S. Nelson-Le Gall, L. Kratzer, E. Jones, and P. DeCooke. 1990. Children's self-assessment of performance and task-related help seeking. *Journal of Experimental Child Psychology* 49, 2: 245–263.

58. Richard S. Newman. 1990. Children's help-seeking in the classroom: The role of motivational factors and attitudes. *Journal of educational psychology* 82, 1: 71.

59. Richard S. Newman. 2000. Social influences on the development of children's adaptive help seeking: The role of parents, teachers, and peers. *Developmental review* 20, 3: 350–404.

60. Eleanor O'Rourke, Kyla Haimovitz, Christy Ballweber, Carol Dweck, and Zoran Popović. 2014. Brain points: a growth mindset incentive structure boosts persistence in an educational game. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 3339–3348.

61. Suporn Pongnumkul, Mira Dontcheva, Wilmot Li, Jue Wang, Lubomir Bourdev, Shai Avidan, and Michael F. Cohen. 2011. Pause-and-play: automatically linking screencast video tutorials with applications. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, 135–144. http://dl.acm.org/citation.cfm?id=2047213

62. Minna Puustinen and Jean-François Rouet. 2009. Learning with new technologies: Help seeking and information searching revisited. *Computers & Education* 53, 4: 1014–1019. https://doi.org/10.1016/j.compedu.2008.07.002

63. Mitchel Resnick, John Maloney, Andrés Monroy-Hernández, Natalie Rusk, Evelyn Eastmond, Karen Brennan, Amon Millner, Eric Rosenbaum, Jay Silver, Brian Silverman, and others. 2009. Scratch: programming for all. *Communications of the ACM* 52, 11: 60–67.

64. Marc Rettig. 1991. Nobody reads documentation. *Commun. ACM* 34, 7: 19–24. https://doi.org/10.1145/105783.105788

65. Glenda Revelle, Allison Druin, Michele Platner, Ben Bederson, Juan Pablo Hourcade, and Lisa Sherman. 2002. A visual search tool for early elementary science students. *Journal of Science Education and Technology* 11, 1: 49–57.

66. John Rieman. 1996. A field study of exploratory learning strategies. *ACM Transactions on Computer-Human Interaction (TOCHI)* 3, 3: 189–218.

67. Allison M. Ryan and Paul R. Pintrich. 1997. "Should I ask for help?" The role of motivation and attitudes in adolescents' help seeking in math class. *Journal of educational psychology* 89, 2: 329.

68. Gina Scala. 2016. How 3D Printing Inspires 8th Graders to Explore STEM Careers. *Stratasys Blog*.

Retrieved September 21, 2016 from http://blog.stratasys.com/2016/02/19/stem-careers-3d-printing/

69. John L. Sherry. 2013. The challenge of audience reception: A developmental model for educational game engagement. *New directions for child and adolescent development* 2013, 139: 11–20.

70. Oliver Stickel, Dominik Hornung, Konstantin Aal, Markus Rohde, and Volker Wulf. 2015. 3D Printing with Marginalized Children—An Exploration in a Palestinian Refugee Camp. In *ECSCW 2015: Proceedings of the 14th European Conference on Computer Supported Cooperative Work, 19-23 September 2015, Oslo, Norway*, 83–102.

71. John Sweller. 1988. Cognitive Load During Problem Solving: Effects on Learning. *Cognitive Science* 12, 2: 257–285. https://doi.org/10.1207/s15516709cog1202_4

72. Rosa Trieu. 2014. What Kids Can Teach Us About Tinkercad And The Future Of 3D Printing. *Forbes*. Retrieved September 21, 2016 from http://www.forbes.com/sites/rosatrieu/2014/09/09/what-kids-can-teach-us-about-tinkercad-and-the-future-of-3d-printing/

73. John Vincent. 2001. The role of visually rich technology in facilitating children's writing. *Journal of Computer Assisted Learning* 17, 3: 242–250.

74. Mark Ward. 2015. How to turn your child into a video games designer. *BBC News*. Retrieved September 21, 2016 from http://www.bbc.com/news/technology-33906050

75. José Pablo Zagal, Anne Marie Piper, and Amy S. Bruckman. 2006. Social and Technical Factors Contributing to Successful 3D Animation Authoring by Kids. https://smartech.gatech.edu/handle/1853/13120