Communicating about computational thinking: understanding affordances of portfolios for assessing high school students’ computational thinking and participation practices

Deborah Fields, Debora Lui, Yasmin Kafai, Gayithri Jayathirtha, Justice Walker & Mia Shaw

To cite this article: Deborah Fields, Debora Lui, Yasmin Kafai, Gayithri Jayathirtha, Justice Walker & Mia Shaw (2021): Communicating about computational thinking: understanding affordances of portfolios for assessing high school students’ computational thinking and participation practices, Computer Science Education, DOI: 10.1080/08993408.2020.1866933

To link to this article: https://doi.org/10.1080/08993408.2020.1866933

Published online: 14 Jan 2021.
Communicating about computational thinking: understanding affordances of portfolios for assessing high school students’ computational thinking and participation practices

Deborah Fields, Debora Lui, Yasmin Kafai, Gayithri Jayathirtha, Justice Walker and Mia Shaw

ABSTRACT

Background and Context: While assessment of computational thinking concepts, practices, and perspectives is at the forefront of K-12 CS education, supporting student communication about computation has received relatively little attention.

Objective: To examine the usability of process-based portfolios for capturing students’ communication about their computational practices regarding the process of making electronic textile projects.

Method: We examined the portfolios of 248 high school students in 15 introductory CS classrooms from largely underserved communities, using a formal rubric (top-down) to code computational communication and an open-coding scheme (bottom-up) to identify computational practices described.

Findings: Students demonstrated stronger abilities to communicate about computation using text than visuals. They also reported under-assessed CT practices like debugging, iterating, and collaborating. Students of experienced e-textile teachers performed substantially better than those with novice e-textile teachers.

Implications: Portfolios provide a viable addition to traditional performance or survey assessments and meet a need to promote communication skills.

Introduction

Computational thinking (CT), defined by Wing (2006) as “solving problems, designing systems, and understanding human behavior, by drawing on the concepts fundamental to computer science” (p. 33), has become a central concept for CS K-12 education, informing frameworks, to national standards, and curriculum design and research (e.g., K-12 CS Framework,” 2016). Yet despite its prominence, researchers and practitioners alike have not only struggled with how to define CT (National Research Council, 2010,
2011), but also how to assess it. For the most part, CT assessments have focused on computational concepts such as logic, algorithms, abstraction, generalization, evaluation, and automation (e.g., Grover, 2017; Grover & Pea, 2018), while a few others have attended to computational perspectives where students express themselves through computing or consider how computing is relevant to their lives. Much less attention has been paid to capturing computational practices such as iterative refinement, testing, and debugging (e.g., Brennan & Resnick, 2012; Grover, 2017; Grover & Pea, 2018; Jayathirtha & Kafai, 2019).

One potential avenue for assessing these under-evaluated computational practices is promoting students’ abilities to communicate about computation. Despite its promotion at the college level as an essential skill in learning how to become a practitioner in the field (e.g., Falkner & Falkner, 2012; French, 2012), communication about computation has received relatively little attention in the K-12 space. Communication about computation involves describing and explaining computational artifacts and related processes and behaviors (College Board, 2017), and can be considered part of the larger category of computational participation (Kafai & Burke, 2014). This practice attends to Conley and Darling-Hammond (2013) broad call for assessments that include communication of ideas through the use of appropriate vocabulary in a subject area as well as presentation of artifacts to a broader audience. From this perspective, communicating about computation is not merely knowing the “vocabulary of computing” (Grover et al., 2014), but also understanding how to use these terms in context in order to fully engage and practice with others as part of the larger “discourse community” of the field (Gee, 1999). Developing this communication competence in computation has been shown to foster “deeper computational learning” and nurture students’ abilities to both think and act upon “computational ideas more effectively” (p. 58) for the purpose of more productive learning. Notably, communicating about computation not only includes conversing with others about computation, but also documenting computational ideas and processes using both text and visual forms that are legible to a knowledgeable audience. Process-based portfolios – more commonly used in arts-based education (Gitomer et al., 1992) – provide a compelling method for practicing this skill. This format has gained traction in CS education as a result of the use of these types of portfolios in AP CS Principles as one aspect of their assessment (Arpaci-Dusseau et al., 2013; College Board, 2017).

In this paper, we report on the use of portfolios in 15 introductory computer science high school classrooms from largely underserved communities in two major metropolitan school districts. The main context of the study was the implementation of the 10–12 week electronic textiles curricular unit (Fields et al., 2018a), an optional end-of-year unit that takes place within the context of Exploring Computer Science (hereafter: ECS, Goode et al., 2012). Electronic textiles (hereafter: e-textiles) connect sewable microcontrollers with conductive thread to actuators such as LEDs as well as sound, light or touch sensors to make interactive craft projects. Mistakes and design changes occur frequently when students make personally relevant e-textile projects, with the implication that there is much to communicate about problems and changes that come up during the process of making e-textiles. For their portfolios, students reported on the function of a final project in the e-textile unit as well as challenges they encountered during the process of making it, namely any problems or design changes they handled. Teachers coached students on how to provide evidence of these problems or design changes, such as annotated images,
to illustrate how they approached those challenges, scaffolding these reflections throughout the unit.

We examined the portfolios of 248 students to address the following research questions: RQ1: When tasked with creating a process-based portfolio about their e-textiles projects, to what extent are students able to communicate about their computational projects and processes? RQ2: What can we learn about students’ engagement with computational practices, especially under-assessed activities such as testing, iteration, and debugging, through these personalized reports on process? RQ3: To what extent does prior teacher experience with implementing computational portfolios to document open-ended design affect the quality of students’ communication about and reporting of their computational experience? To answer these questions, we used a rubric to evaluate the quality of students’ communication quality and conducted an open-coding process to identify the range of computational thinking practices evidenced in students’ portfolios. Finally, we compared the portfolios of students taught by experienced e-textile teachers (three classrooms) to those of students taught by novice e-textile teachers (twelve classrooms) both in terms of their communication quality and reports on CT practices. In the discussion, we address affordances of using portfolios to assess students’ abilities to communicate about computation, as well as their engagement of computational practices in CS classrooms.

**Background**

One major challenge in assessing computational thinking is evaluating students’ depth of understanding. Conley and Darling-Hammond (2013) argue that beyond core subject matter concepts, assessments must measure students’ higher-order cognitive skills, skills that support transferable learning and also abilities like complex problem-solving, planning, reflection, collaboration and communication. For this reason, Grover (2017) proposes a “system of assessments”, or multiple complementary measures that provide an expansive picture of students’ learning, including cognition, affect, and transfer ability, whether evaluation of student-generated programs, specially constrained design tasks, or isolated debugging tasks (e.g., Fields et al., 2016; Moreno-Leon et al., 2015; Werner et al., 2015). Most commonly, computational assessments focus on student comprehension of computational concepts, using tools such as carefully designed quizzes, multiple choice tests, special coding tasks (like Parsons problems), or artifact-based interviews (e.g., Barron et al., 2002; Brennan & Resnick, 2012; Cooper et al., 2010; Grover, 2017; Parsons & Haden, 2006). Less common are assessments that focus primarily on computational practices and process, including active observation of students’ problem solving, think aloud studies of students’ debugging or design processes, or interviews about the kinds of reasoning students apply (e.g., Jayathitha et al., 2020; Lewis, 2012). Yet amidst all of these, certain practices that Conley and Darling-Hammond (2013) call for in assessments, such as reflection, collaboration, and in particular, communication are missing. To this end, we focus on process-based portfolios for two reasons: first, as a means of facilitating student reflection and communication about computation, and second, as a a means of capturing student engagement with under-assessed
computational practices such as testing and iteration, debugging, and isolation of problems.

**Portfolios as assessments**

Portfolios have been used across many disciplines and fields for assessment, though their actual formats and contents vary widely, each with different affordances for assessing student learning. One predominant type of portfolio, the “cumulative” portfolio, focuses on assembling a series of finished projects. Popular in art and design, this portfolio format involves curation of one’s best projects over time, allowing a “showcase” of overall competency and skill (Býrgýn & Adnan, 2007). In contrast, portfolio formats derived from writing (Williams, 2002) tend to focus more on continuous documentation of students’ learning and growth (Paulson et al., 1991). For instance, the “process” portfolio highlights the history of a single project, illustrating the trajectory from ideation to construction (McKay et al., 2015), while the “reflective” portfolio centers on an individual’s growth or learning over a particular period of time, such as a semester or across a program (Paulson et al., 1991). These types of portfolios focus on building a narrative around one’s own progress and growth, and are the basis of the portfolio assessment designed for this study.

Portfolios used in science, technology, engineering, and mathematics (STEM) domains tend to be more process-driven and reflection-focused, often dealing with the progression of a single project (Chang et al., 2015). Rather than only presenting a final project, these process-based portfolios often include a collection of different kinds of in-progress evidence compiled over a student’s trajectory. This might include a range of non-textual forms of communication, such as initial prototype sketches for engineering projects (Eris, 2006), flowcharts and code throughout the development process (Higgs & Sabin, 2005), or photos of projects in process (Chang et al., 2015). Students provide textual explanations for this evidence, whether explanatory captions and annotations (Býrgýn & Adnan, 2007), articulation of underlying concepts (Phelps et al., 1997), or narrations of growth and learning (Paulson et al., 1991). These process-based portfolios offer insights into students’ processes and thinking because of their situation within everyday practice, since it requires that students keep track of their ongoing work along the way (Býrgýn & Adnan, 2007).

As such, process-based portfolios are much better able to capture a more holistic view of student understanding and learning because they focus on process alongside product (Paulson et al., 1991). Here, learning and assessment can become seamlessly integrated within classroom practice (Gilman et al., 1995). For students, creating a portfolio can also provide agency in shaping one’s learning over time, whether through continuous self-feedback and progress monitoring (Adams, 1998; De Fina, 1992), or purposeful opportunities for goal setting (Owings & Follo, 1992). Further, these types of portfolios highlight the actual aspects of creation and reasoning around production that are normally hidden when relying primarily on final products for assessment. In computer science, process-based, reflective portfolios can build on existing reflection and communication practices already used in some courses. For example, the practice of keeping a design notebook has already been noted as a way to become enculturated into fields of computer science and engineering in higher
education (Eris, 2006). Similarly, reflective journal entries are a key practice in the equity-driven K-12 Exploring Computer Science curriculum as a way for students to not only share their work with others, but deepen their understanding of computation (Goode et al., 2012). Building on these reflective practices, a process-based portfolio can provide a focused means for communicating one’s ongoing trajectory of learning and understanding through unpacking the process of creating a computational artifact.

**Portfolios to support communication about computation**

Within CS classes, portfolios offer the potential for supporting students’ communication abilities, specifically their capacity to describe and explain computational artifacts and related processes and behaviors (College Board, 2017). While there have been some efforts to teach communication skills within CS courses at the university level (e.g., Falkner & Falkner, 2012; French, 2012), the most predominant effort to support computational communication in high school education is through the portfolio requirement for the recently launched AP Computer Science Principles (AP CSP) course. In the course, a portfolio supplements the standard multiple-choice exam, providing students an opportunity to show the development of their own algorithms and programmed projects. Here, they answer questions about a computational innovation they created, the tools and processes they used to create it, and the potential benefit and harm of how their creation consumes and produces data (Arpaci-Dusseau et al., 2013; College Board, 2017). The way that students are expected to accomplish this includes leveraging “accurate and precise language, notations, or visualizations” (College Board, 2017, p. 10) – in other words, focusing on both text and images. This dual focus in communication was part of the inspiration for our own requirement of textual and visual communication in our portfolio design.

Textual communication of computational thinking is not just a matter of vocabulary (i.e., knowing the right words to describe specific computational concepts), but also how well students can articulate and accurately capture their computational thinking practices through text and other available media (Grover & Pea, 2018). More concretely, this means that students’ communication about computation can fall into different levels of written competence (Taffe, 1989). At the first level, this might encompass using a shared discourse and vocabulary of computing, such that they know the appropriate terms to describe specific ideas or procedures within the field. At the second level, students should have the capacity to employ this language to help explain larger computational systems or projects. Finally, the highest level of communicative competence emphasizes students’ ability to clarify, support, and potentially defend their original computational ideas or projects to others. These features make communication a key illustration of computational participation (Kafai & Burke, 2014), which highlights the underlying social and cultural dimensions in understanding and designing computational systems where learners can effectively communicate and connect with the broader community about their creations.

Generating appropriate visual representations is another key aspect of becoming a competent computational communicator. Here, the AP CSP guidelines include not only “written and oral descriptions” as part of communication, but also how these
designations are “supported by graphs, visualizations, and computational analysis” (College Board, 2017, p. 10). As part of their performance task assessments, they require students to create several visual aids, including annotated excerpts of code, and an original computational artifact that illustrates or represents their ideas in a “nontextual” way. Thus, beyond mere presentation of images, visual communication of computation encompasses creating and using visual evidence to further support a narrative and/or progress an argument. In general, the ability to not only interpret, but also create visualizations speaks to research that highlights the importance of doing so in order to promote greater scientific reasoning, create alternate opportunities to learn, and join the larger scientific discourse community (Ainsworth et al., 2011). Communicating effectively with visual evidence may be particularly relevant when thinking about physical computation, such as e-textiles and robotics, where fabricated objects are multidimensional and require multiple modes to convey all the elements of a project (McKay et al., 2015). Thus we chose to emphasize both textual and visual communication in the process-based portfolio we designed for the e-textile curricular unit.

**Designing portfolios for computer science classes**

The instances of portfolio uses in undergraduate engineering courses and high school AP CSP exams documented above inspired us to consider portfolios for assessing students’ computational thinking within classroom settings to capture more robust insights into students’ achievements, as well as situating assessment in more authentic, real-world contexts. Moreover, in the context of the e-textiles unit within the ECS curriculum, students not only program, but also design circuits, and construct physical textile artifacts – dimensions that are difficult to capture in a traditional written exam (for an exception, see Litts et al., 2017). A recent review of the implementation of e-textiles during the last decade revealed that while e-textiles activities were successful in broadening students’ access and increasing interest in CS, there was little evidence of what they accomplished in deepening students’ learning of computational concepts and practices (Jayathirtha & Kafai, 2019). Therefore, a process-based portfolio could allow students to describe their personal process, in their own words, and thus provide deeper insights into their computational understanding and practices (Brennan & Resnick, 2012; Shaw et al., 2019).

Over the course of several years, we designed, iterated, and implemented three different digital portfolio formats that were intentionally process-based. In our first effort to analyze the usefulness of portfolio assessment for computation, our participating teacher designed her own portfolio format (implemented in one classroom). This portfolio, created using an iBook format, included four sections, each focused on a domain particular to e-textiles: design, circuitry, crafting, and coding (Lui et al., 2019). While this portfolio format was successful at capturing a general sense of students’ engagement with CT practices (e.g., that students debugged their code), often these descriptions were vague and did not provide enough detail to link these activities to specific computational concepts or ideas (e.g., what changes were actually made and why). Additionally, while students provided a plethora of visual evidence (e.g., code and pictures), these often lacked adequate explanation or contextualization within the narrative of the portfolio, thereby making it difficult to get a full picture of their computational understanding.
In the second version of portfolios (implemented in three classrooms), we encouraged both the reporting of process and self-reflection more explicitly, requiring students to create three sections for a digital portfolio presentation (Google Slides). These included: 1) *Product* – the final appearance and functionality of the project; 2) *Process* – any debugging or revisions that occurred during creation; and 3) *Reflection* – what they learned from the entire e-textile ECS unit. Students were also required to provide visual evidence to support these three categories, though they had the choice of what to include, whether blueprint drawings, screenshot of their code, or pictures of their projects at different stages of completion. As a result, in this iteration, students created their portfolios in personally relevant ways that acted as rich ideational resources to reflect on and construct their CS identities (Shaw et al., 2019). However, communication of their actual computational knowledge and processes were inconsistent (Lui et al., 2018). While some students were able to report about one or two of their debugging episodes with great detail, others provided over-generalized descriptions trying to encompass the entire scope of their process (e.g., resewing a part of their project, or double checking their code multiple times). Here, it was difficult to get a sense of how their computational knowledge informed these changes. Further, students’ visual communication was still ineffective at supporting their narratives and arguments. As in the prior study, all students included multiple images and visualizations within their portfolios, but these were often missing annotations or captions to clarify their relationship to the textual descriptions or topic at hand.

Keeping in mind the findings from these two pilot studies, we redesigned the portfolio for the large-scale implementation of the e-textile unit reported in this study to better scaffold students in creating more effective descriptions and visualizations of abstract computational concepts and processes paper. By doing so, we were not only interested in improving the clarity of students’ communication about computation but also working to increase reports about their computational practices in general. For the third version of portfolios, (i.e., the focus of this study) students again created digital portfolios (format was the teacher’s choice) at the end of the e-textile unit to highlight their final project (implemented in 15 classrooms). The portfolio had the same three required sections as the previous iteration (*Product, Process, and Reflection*), but added more limitations in the latter two sections in order to clarify students’ communication. Rather than allowing students to choose the scope of their own descriptions, we specifically asked them to detail only *two* challenges faced or changes made (including at least one that related to coding), and *one* thing they learned through the unit. Also, we explicitly outlined what an effective description should include, for example, both a problem *and* how it was resolved, or a before *and* after image of a revision (see Table 1). In providing these guidelines, we aimed to help students in producing more effective reports about their computational processes.

Further, we provided a grading rubric to support clear expectations for good computational communication, which included separate categories for textual and visual communication. Within the curriculum itself, we additionally added structured supports for portfolios, acknowledging the importance of teacher practice in scaffolding students’ learning. This included two exemplar portfolios (created by the research team), and several student portfolios of varying quality from the prior year, with classroom discussion prompts on elements of effective visual communication (e.g., using visual markers,
Table 1. Summary of textual and visual requirements of the portfolio assignment. The points allotted to each section for the portfolio grading rubric are also included below.

<table>
<thead>
<tr>
<th>Sections (100 total points)</th>
<th>Text-based Requirements (60 points total)</th>
<th>Visual-based Requirements (40 points total)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description of Project</td>
<td>Documentation of final projects including its aesthetic design, physical form, and coded behaviors. (15 points)</td>
<td>Photos of the project including placement of all the electrical components (LEDs, sensors, and microcontroller). (10 points)</td>
</tr>
<tr>
<td>(25 total points)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Description of Process</td>
<td>Reporting on two “process moments” including either: (30 points)</td>
<td>For each “process moment”, including a before/after image, whether code screenshots, photos/drawings of the project, circuit diagrams, or engineering notebook or journal entries. (20 points)</td>
</tr>
<tr>
<td>(50 total points)</td>
<td>• two revisions made (changes and justifications);</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• two challenges faced (problems and solutions), or</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• one revision and one challenge.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>At least one of these moments must focus on coding/programming specifically.</td>
<td></td>
</tr>
<tr>
<td>Reflection on Learning</td>
<td>Discussion on one specific skill or competency gained throughout the entire e-textiles unit. (15 points)</td>
<td>Any kind of photos, drawings, or otherwise to help illustrate the skill and competency gained. Images from multiple projects allowed. (10 points)</td>
</tr>
<tr>
<td>(25 total points)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

captioning pictures, annotating code). In this way, students could analyze portfolios of varying quality in order to come to their own conclusions about what count as productive forms of visual and textual communication. As part of classroom activities, we also added several design notebook prompts (already a key part of the curriculum) to support students’ documentation of specific errors fixed or changes made in the course of their designs. This would provide students with in-the-moment records of challenges faced or changes made, which could then be drawn upon during the process of creating their portfolios at the end of the unit. During the four-day teacher PD for the e-textiles unit, a half-day was devoted to portfolios and the portfolio rubric, where teachers evaluated prior student work and discussed strategies for supporting clear and specific computational communication through text and visuals.

**Methods**

**Context and curriculum**

This study is part of a larger, three-year effort to develop and research the electronic textiles unit of *Exploring Computer Science* (ECS). Three equity issues – broadening access, diversifying representation (by privileging non-dominant CS learners, practices, and artifacts), and deepening participation – formed the impetus for developing the e-textiles curriculum (Fields et al., 2018a). The 10–12 week e-textile unit centers around the design of four e-textile projects that build in complexity and creative freedom, helping students learn challenging concepts in computing, electronics, and crafting while also supporting personal expression and design (for design principles of the unit, see Fields et al., 2018a; for in depth descriptions of all projects see, 2018b; the unit in its final form is available at [http://exploringcs.org/e-textiles](http://exploringcs.org/e-textiles)). The final two projects of the unit are the most computationally complex. The third project consists of a collaborative classroom mural where
pairs of students create individual pieces with a collectively chosen theme. Each piece incorporates four independently circuited LEDs sewn onto a piece of felt, utilizing two built-in switches on the Adafruit Circuit Playground microcontroller to trigger four lighting patterns. The fourth project incorporates a handmade human sensor, attached to either an existing object such as a sweatshirt or a newly created object such as a plush toy. The sensor consists of two aluminum foil conductive patches that when squeezed generate a range of data, which are programmed to correspond to different lighting effects based on how hard a user squeezed the project. The curriculum includes a 100-point rubric for the final project, equal to the number of available points for the portfolio, in order to support product and process equally as part of the system of assessments (Grover, 2017) of this e-textile unit.

**Participants**

In Spring 2017, the ECS-school district liaison in a large metropolitan area in California sent an invitation to the ECS-teacher listserv for educators to participate in the third and final year of the e-textiles pilot study. Teachers and school sites were chosen by the liaison for their diverse teaching styles, to maximize the variety of feedback on the curriculum. The liaison included himself as a pilot teacher. Lastly, when two teachers announced their need to take maternity leave mid-pilot, the replacement teachers were identified through snowball sampling. The final number of educators was 15, with teaching experience ranging from 3 and 37 years; most but not all also had taught ECS for several years. Twelve teachers were brand new to the e-textile unit (hereafter: “novice e-textile teachers”) and three teachers had taught the e-textile unit the prior year (hereafter: “experienced e-textile teachers”). All teachers engaged in four days of e-textile professional development over a period of four months, where they became familiar with the curriculum by creating the projects from the unit and reflecting on the pedagogy used during professional development.

In Spring 2018, these 15 teachers implemented the e-textile unit in their ECS classes; classes ranged from 20–42 students. For participating schools, percentages of students from ethnically underrepresented groups (i.e., non-white) ranged between 72% and 99%; English as a second language between 2% and 41%; and those with free or reduced lunch between 47% and 97%. Over 430 students participated, but only 359 provided consent to collect their data for research purposes. Of these, 272 completed both the pre- and post-surveys of our larger study about the e-textiles curriculum (described in Kafai et al., 2019). For this study, we collected all available portfolios, ranging from 2–35 per teacher, and ended up with portfolios from 248 students available for analysis. Key reasons for the variable rate of portfolio return are discussed below.

Teachers differed in terms of how they implemented the portfolio assignment in their classrooms. This is part of the variation that teachers demonstrated in implementing the curriculum itself (Shaw et al., 2020). The main variation that influenced the portfolio return rate was the scheduling of the unit itself. While all teachers introduced portfolios as a part of the unit, some ended up not requiring submission of portfolios based on their own time constraints. Mostly, this was based on when teachers themselves decided when to start the unit (January – March 2018), how long they spent on it (10–13 weeks), and how many projects they covered in class. Five teachers (all novice e-textile teachers) faced
a lack of time at the end of the school year and decided to do only three out of the four projects, focusing portfolios on the third (mural) project rather than the fourth (human sensor) project. Two other novice e-textile teachers allowed students to create portfolios collaboratively, whether in pairs (especially for the mural project, a pair-based project by design) or in groups of four (in the case of one teacher who, stretched for time, had students work on the human sensor project in pairs and the portfolio in fours). In classrooms where not all students submitted portfolios, teachers were focused on allowing students as much time as possible to finish their projects.

One important thing to note here about portfolio return rate is that the issues described above only occurred with novice e-textile teachers, who were teaching this unit and content for the first time. The experienced e-textile teachers, who had already taught this e-textiles unit and implemented the portfolio assignment (second iteration) in the year prior (for more information on this implementation, see Lui et al., 2018), were successfully able to complete the unit (i.e., all four projects as outlined), with nearly all of their students submitting portfolios. As will be described later, this difference is mostly attributable to prior experience with managing time for projects versus portfolios, as well as actively working to support portfolio creation throughout the design and construction process of e-textiles projects itself. Potential challenges of implementing portfolios in the classroom, as well as suggested future areas of research, are further addressed in our discussion.

**Portfolio rubric**

The portfolio rubric supports evaluation of students’ performance on the digital portfolio. This rubric was organized into the same major sections as the portfolio: Product (25 points), Process (50 points), and Reflection (25 points). For each of these sections, students were evaluated on both: (1) textual description quality – inclusion of required information and relevant details, as well as clarity of language (60% of the score), and (2) visual representation quality – inclusion of relevant photos, drawings, or code excerpts, and if these were clear and easy to decipher through use of accompanying explanations, annotations/markers, or captions (40% of the score) (see Table 1 for point distribution). In order to support consistency of evaluation, we provided four-tier score categories for each subsection (e.g., for the 15 possible points for the textual description of the project, four categories of 0, 5, 10, or 15 points) with clearly laid out specifications for each category in terms of required content, description, and visuals. The full portfolio rubric is available in the e-textile unit and in the online supplemental materials (the first section of the portfolio rubric is in Figure 1). Below we describe the larger context of the e-textile unit within Exploring Computer Science and the participants and context of this particular study.

**Data collection**

We collected two forms of data including: the digital portfolios themselves and portfolio rubric evaluation scores. For the portfolios we collected, we removed all non-consenting students’ portfolios, and a handful of (unopenable) corrupted portfolio files and incomplete portfolios (indicating that students started but did not complete the final portfolio).
This left 221 portfolios from 15 classrooms, which included 204 individual portfolios and 17 group portfolios (for n = 248 students). As described above, teachers varied in terms of how they implemented the portfolios. While most teachers used the suggested digital presentation format (i.e., Google Slides, Microsoft Powerpoint) (81.4% of portfolios), some focused on alternate formats including multi-page websites (13.1%), digital text documents (i.e., Google Documents, Microsoft Word) (3.2%), and digital presentation with accompanying audio narration (i.e., a video) (2.3%). Additionally, some novice teachers focused on the pair mural project for the portfolios rather than the individual final human sensor project because of time constraints described above. For all portfolios, we tabulated their different digital formats provided and details about the projects described, which are all summarized in Table 2 below.

For portfolio rubric evaluations, we collected all available portfolio rubric evaluations from teachers. For consistency, we kept rubric evaluations only for portfolios for which we had a copy that would enable verification of grading and coding. Complete teacher-generated scores with accompanying section subscores were available for 7 of 15 classrooms (n = 99 portfolios representing 44.8% of all portfolios examined). As mentioned earlier, teachers received training on using the rubrics and evaluating sample student projects during PD. However, as with implementing the portfolio assignment, teachers varied in terms of how they graded these assignments. Some teachers used the rubric in ways that did not support data collection. Two teachers provided only the completed scores (out of 100) with no accompanying subscores (for the three sections, or text vs. visual communication), while one teacher only provided partial scores generated through peer grading in his class (a regular part of his pedagogical practice). In addition, five

<table>
<thead>
<tr>
<th>Medium of Portfolio</th>
<th>E-Textile Project</th>
<th>Form of Object</th>
<th>Genre of Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Digital Slide Presentation: 81.4%</td>
<td>Human Sensor Project: 79.2%</td>
<td>Existing Artifact (e.g., a t-shirt bought at a store): 40.3%</td>
<td>Plush Toy: 49.3%</td>
</tr>
<tr>
<td>Webpage: 13.1%</td>
<td>Mural Project: 20.8%</td>
<td>Original Artifact (e.g., handmade stuffed animal): 51.6%</td>
<td>Flat Object (e.g., banner, wall-hanging): 26.7%</td>
</tr>
<tr>
<td>Text Document: 3.2%</td>
<td></td>
<td>Unknown: 8.1%</td>
<td>Clothing: 13.6%</td>
</tr>
<tr>
<td>Narrated Slides (voice-over video): 2.3%</td>
<td></td>
<td></td>
<td>Accessory (e.g., hat): 8.1%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Unknown: 2.3%</td>
</tr>
</tbody>
</table>
(novice) teachers provided no scores at all because of the time constraints described earlier. As such, one member of our research team (Author1) used the rubric to grade the portfolios from those classrooms \((n = 122)\) portfolios representing 55.2\% of all portfolios examined). Again, as with portfolio collection, potential teacher implementation issues regarding rubric evaluations and suggested areas for future research are further described in our discussion.

**Data analysis**

For this analysis, we implemented a mixed-methods approach. In order to answer RQ1 regarding the quality of students’ computational communication skills in submitted portfolios, we looked at the grading rubric scores of student portfolios as evaluated by the teachers and the first author, calculating descriptive statistics of the different score categories for each portfolio. In order to answer RQ2 regarding students’ reports of their engagement with computational practices, we looked at the content of digital portfolios. Here, we developed a grounded coding scheme through a two-step, open-coding process (Charmaz, 2006). In the first step, authors one and two drew on existing sources in order to develop an initial coding scheme. One source we drew from was Brennan and Resnick (2012) discussion of computational thinking practices (including being incremental and iterative, testing and debugging), and computational thinking perspectives (personal expression, connecting with others, and questioning one’s environment). We also drew from our prior analysis on portfolios from year 2 of the project (Lui et al., 2018; Shaw et al., 2019), where we considered what students were most likely to report upon in terms of their process and learning, especially regarding collaboration, aesthetics, and personal learning.

The two researchers applied to these initially generated coding categories to 15 randomly chosen portfolios from the three classrooms where teachers had prior experience with e-textiles (5.9\% of students). In particular, we focused on the two latter sections of the portfolio *(Process and Reflection)*, which emphasized students’ experiences constructing their projects, and their self-reported descriptions of what they learned. We then wrote analytical memos (Saldana, 2011) about this initial application of these categories, which we used to further refine categories until we created our final coding scheme (see online supplemental materials), for instance, clarifying what subcategories should fall under the larger category “Design Changes”. We reapplied the final new scheme to the 15 originally selected portfolios until we established 100\% consensus between the first two coders. We finally trained an additional three coders (Author4, Author5, and Author6) on the new coding scheme using the same set of portfolios, in preparation for coding all the available portfolios.

In order to reach inter-rater reliability, the five coders then jointly coded a randomly selected set of portfolios (9.8\% – 20 of 221 portfolios across all classrooms) where we established strong inter-rater agreement with an intraclass correlation coefficient = 0.75; \(p < .001\) (a correlation of 0.81 and above is nearly perfect and above 0.70 is considered acceptable). Among the five researchers, we divided and coded the remaining portfolios \((n = 201)\). In the analysis, we calculated the number and percent of portfolios within each of the codes described above to help judge how predominant the codes were across students’ portfolios.
Finally, for RQ3 regarding a comparison between portfolios of students taught by teachers with and without prior experience with e-textiles, we compared the findings for both RQ1 (rubric scores) and RQ2 (coding for computational practices) between classes taught by novice e-textiles teachers (12 teachers, teaching e-textiles for the first time) versus classes taught by experienced e-textile teachers (3 teachers, teaching e-textiles for the second or third time).

**Findings**

Our findings are divided into two sections. First, we report on the quality of student communication about computation seen in the submitted portfolios using the rubric evaluations (RQ1). Second, we report on what students’ shared about their computational thinking practices within the portfolios (RQ2). For both sections, we compare the findings between students who were taught by novice e-textiles teachers (teachers who were teaching the unit for the first time) versus students taught by experienced e-textile teachers (the three teachers who piloted the unit for 1–2 years previously) (RQ3).

**Student communication about computation**

Overall, students who submitted portfolios demonstrated a solid level of competence in communicating about their e-textile product, process, and learning. For all 15 classrooms, the scores ranged from 3 to 100 (out of 100), with a mean of 75.0, median of 75, a mode of 100, and a standard deviation of 24.2. Except for the few students who purposefully submitted incomplete portfolios (but likely prioritized finishing the projects themselves), we consider scores above 70 very high. The summary of rubric scores are shown in Table 3.

Notably, the overall portfolio rubric scores were generally higher for students who had experienced e-textiles teachers, with a range of 35 to 100, a mean of 85.9 and a standard deviation of 14.67. This tendency toward higher scores was present throughout all sections of the portfolio, for which may have occurred for several reasons. First, the experienced teachers all made enough time to complete the entire e-textile unit and allowed sufficient time for students to create their portfolios. Further, all three experienced e-textile teachers already had one year of experience implementing the portfolio as part of the curriculum, and themselves provided feedback that led to the design changes between the second and third iterations.

<table>
<thead>
<tr>
<th>Table 3. Average rubric scores across portfolio sections by textual and visual communication.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Portfolio Section</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Project 88.7%</td>
</tr>
<tr>
<td>Process 87.8%</td>
</tr>
<tr>
<td>Reflection on Learning</td>
</tr>
<tr>
<td>OVERALL</td>
</tr>
</tbody>
</table>
of the portfolio (reported in the background section of this paper and in Lui et al., 2018). The comparison between the rubric scores between experienced, novice, and all e-textiles teachers is detailed in Table 3.

Textual computational communication

In terms of the students’ textual communication performance, students generally scored the highest within the Product section (mean 84.2%), where they were asked to describe their final project. This may have occurred because this section required factual, concrete descriptions of the structural and functional aspects of students’ projects, including where the LEDs and sensors were placed, as well as what their light patterns did (e.g., LEDs blinking back and forth slowly). This can be seen in Fernando’s description of his project illustrates a strong textual project description:

In this project what we did was that we made a 3D figure of something that could’ve been a pillow, teddy bear backpack, etc. what i went with was a pillow, the pillow had to have three LED lights and two patches that when touched with a specific amount of pressure the pattern of the lights would change. On the pillow i had a dinosaur figure and the 3 lights were placed in a row on the tail, the patches were placed to the side. I had 4 patterns of the lights

1. Pattern 1 show us 3 lights taking turns turning on and off
2. Pattern 2 each light rapidly blinks at its own time
3. Pattern 3 two lights turned on at the same time while the other was off then they switched
4. Pattern 4 all lights slowly turned on in a sequence

Here, Fernando precisely outlined the details of both his physical project form and light patterns that he coded, even describing the choice he made in creating a pillow instead of a different form of project. Each lighting pattern is clearly described within a numbered list. Because of his inclusion of all the required details, as well his clarity of description, Fernando received full credit for this section of the rubric.

Students scored slightly lower in textual communication in the remaining two sections of the portfolio: Process (mean 75.4%) where students had to describe their challenges faced and/or revisions made, and Reflection (mean 79.0%), where they described their growth and learning. There are several reasons why this may have occurred. First, students may simply struggle more with describing processes more than products. Describing a process moment requires thinking across time and being aware of the reasons for one’s actions. Second, process moments took place while projects were being made and often before the creation of the portfolio itself; unless students had documentation (such as design notebook entries) about mistakes or revisions they made, it may have been more difficult for them to remember details. Third, many students who provided high levels of detail occasionally lost points because they focused only on one process moment instead of the required two. This makes sense considering the fact that even describing a single moment by providing complete details and clarification required more effort and space than vaguely describing two moments.

Below are two examples of students’ textual descriptions of process moments that illustrate the range of detail that students included. Here, both students wrote about dealing with sewing/crafting and related circuitry issues in their projects. While both
students use the span of a single paragraph, Alma described two process moments with vague and general language. Dhruv, on the other hand, described one sewing issue in detail within a single paragraph:

My first obstacle was the outline of my project. My problem was that they [conductive thread lines] crossed with one another. I tried a couple of times (which wasn’t a success), then tried a different setup and worked perfectly. My next challenge was my sewing. The first time I sowed I had messed up pretty badly. Only 3 of my lights were turning on so decided to re-sow. Which then ended up working (Alma, p. 4)

When I had begun sewing my final project, I faced an obstacle. Both the foils [touch sensors], negative and positive, were not reading the amounts of pressure applied to the project to change the light patterns. I tried to redo the foil to create more connectivity from the foil to the conductive thread. I redid the foil and chose a much bigger piece this time. Instead of sewing all over the foil, I decided to go around the whole foil but around on the edges. This made a huge difference, and I was receiving much better results from the foil immediately. The project is now able to read all the four different light patterns depending on the amount of pressure applied on the foils. (Dhruv, p. 3)

In the above descriptions, both students reported on their problems and subsequent solutions, though to differing degrees of detail. While Alma describes two problems in this excerpt, both descriptions lack specific details about the actual nature of the problem (e.g., “The first time I sowed I had messed up pretty badly. Only 3 of my lights were turning on”) and its solution (e.g., “I decided to re-sow. Which then ended up working”). Dhruv, on the other hand, provided not only more details about what his problems were (“Both the foils [touch sensors], negative and positive, were not reading the amounts of pressure applied to the project to change the light patterns”), but also outlined two major changes he made to resolve it: larger foil patches and better sewn connectivity across a greater area of the patches. Notably, Dhruv’s second challenge report (not included here, but also a requirement of the portfolio) was just as highly detailed as the paragraph above. Some students received less credit for their process moments because they only included one moment, even if it was described in great detail. This reveals some of the limitations of the portfolio guidelines and rubric, which did not provide as much information on the level of detail in students’ textual explanations as long as they included an issue and the resolution.

In the Reflection section, many students were able to communicate textually about the concrete hard skills and competencies they acquired (e.g., sewing, programming). However, these descriptions were generally less clear and detailed when writing about abstract ideas such as improved soft skills (e.g., creativity, teamwork, planning). While students mentioned their soft skills, it was often difficult to judge whether or not they could provide the requisite evidence for the development of these skills. For instance, David chose to describe improvement in sewing skills, which he demonstrated with three pictures and explained textually:

Through the E-textiles unit, I have improved on one skill drastically, and that skill is sewing … [proceeds to show three images that highlight growth in sewing ability across different projects] As you can see, my sewing skills improved drastically throughout these projects.
They went from a huge mess to clean stitches with cleaner cut offs. What I didn’t take into consideration in the beginning that I think made my sewing look bad is which side I would [tie and] cut the sewing off. On the first project, I cut the thread off on the front, so that made it look bad. (David, p. 5)

Here, we can see how David was able to include concrete details (“clean stitches with cleaner cut offs … ” and knowing “which side I would [tie and] cut the sewing off”) to demonstrate his growing ability to sew an e-textiles project. Another student, Klinge, had more difficulty describing how his design skills had improved:

One thing I improved on in the e-textiles is designing. I learned how to make design and follow through with them. Designing really helped me get to my final product. It was difficult designing it and choosing which port the conductive thread would go to. (Klinge, p. 7)

Klinge describes that he improved on “designing” but provides little detail about how that design actually changed besides the fact that he was able “follow through with them”. While he mentions the act of figuring out the circuit connections of the project (“choosing which port the conductive thread would go to”), he does not detail whether or not this is an aspect of the design, or through what means this activity became easier for him (e.g., changing the spatial arrangement of components, deciding upon the grounding line for components first).

This challenge of providing details for less concrete skills speaks to the need to further scaffold students when reflecting on developing abstract or soft skills within future portfolio designs. Students may need help defining what an improvement in collaboration, design, planning, and similar soft skills looks like in detail. Further, as we discuss below, these abstract soft skills proved particularly difficult when thinking about visual communication as well.

**Visual computational communication**

Students who submitted portfolios demonstrated a basic level of competency with visual communication, albeit with a much lower score than on textual communication (mean: 59.5%). As with textual descriptions, students overall scored much higher in the *Product* section (mean 74.6%). As with the textual communication, this is likely due to the concrete nature of reporting on the final project itself, and to the fact that images could easily be gathered at the last minute to support their descriptions. Students here featured final pictures of their projects, and many added annotations such as arrows or captions to help explain the placement of components. As an example, Kristin provided two images in this section, which included an overall view large enough to see the placement of the patches on a plushie doll as well as a close-up figure of the circuitry of the lights in the doll’s heart (See Figure 2). Note that her visuals are carefully annotated with arrows and labels pointing to the lights and an explanation for why she needed two diagrams.

Within the *Process* and *Reflection* section, students scored much lower for visual communication (55.6% and 52.4% means, respectively). Here, some students did not annotate or label their evidence, while others provided no visuals at all. As with the textual communication, one potential reason why students received lower scores in the process section was because of timing issues. When dealing with issues during construction, many students were likely more involved in actually making their projects work, rather than stopping to visually document their mistakes, such as with screenshots of
errors in their code, or pictures of messy sewing that caused short circuits. Notably, this lack of recorded evidence is something that some students explicitly mentioned within their portfolio (e.g., "No pictures of this").

To demonstrate the range of annotation that students provided for visual evidence, we share three examples. Nino’s explanation of fixing an error in his code demonstrated what effectively annotated visual evidence can look like (see Figure 3). Alongside a substantial textual explanation of the change he made in coding the ranges of his sensor conditions, Nino provided before and after screenshots of his code with detailed captions that explain exactly what changes he made. Using a different but similarly effectively method, Angelica provides an image annotated with arrows and labels (alongside a detailed textual description) to compare changes she made within her circuit diagram (see Figure 4). In contrast, Cesar also provided multiple images to accompany his process moment description, however, he does not annotate or caption these to aid in supporting his descriptions. Here, it is unclear how these pictures relate to the two challenges he describes ("Figuring out the coding" and "getting the wires to fit on the bat") since they do not show his excerpts of his code, nor highlight the stitching on the project (see Figure 5).

The vast range of performance of students on visual annotation of process moments illustrates the need to further support students with documenting (as well as solving) these moments of troubleshooting and debugging. The differences between students who were taught by an experienced versus novice e-textile teachers is particularly pronounced in this area (87.8% versus 49.2%, respectively). This suggests that the experienced teachers may

---

Figure 2. Kristin’s visual explanation of the design and circuitry of her human sensor project (p. 3).
have helped their students better prepare for reporting on these process moments better during the weeks of project creation including the suggested scaffolds mentioned in the methods section (e.g., in-class journal prompts, use of exemplars, class conversations).

Generally, students also scored lower for visual communication in the Reflection section. This parallels trends within textual communication for this part of the portfolio, especially when thinking about students’ reporting on hard versus soft skills. For example, consider John’s description of how his sewing improved (see Figure 6) where he compared the sewing of his first project (in yellow, on left) to the sewing on his final project (in black, on right). This image accompanied a textual explanation describing how he learned to sew smaller stitches that conducted electricity better (because the thread is more taut). Thus, students who reported on hard skills (e.g., sewing, drawing) could more easily present visual evidence of their improvement because it was more concrete. Students
who reported on soft, abstract skills (e.g., planning, patience) tended to present less images in general to support their narrative. This difference can be seen in Zoe’s portfolio, where she explained:

Sure I learned how to sew, but what’s most important to me that I know I learned is patience. Patience was a huge key involving this unit, not only in computer science, but patience has helped me throughout many classes this year. And so, without patience, you cannot really accomplish anything in life. (p. 6)

Here, Zoe does not provide any images to support her growth in patience, and even comments that she could have focused on sewing as an area of growth (which might
On the left is an example of my sewing from the very first project compared to the image to the right displaying my sewing ability from this project.

Figure 6. John’s picture and annotation comparing sewing in his first and last projects, demonstrating growth through smaller stitches.
have better lent itself to visual documentation) but chose not to. It is difficult to imagine what kind of imagery could evidence her growth in patience here, an area that she claims has influence on other scholastic areas of her life. This speaks to the need for us to reconsider how either form of communication (textual or visual) might not always be the best way of communicating particular types of computational information, and how to appropriately support students in choosing the most effective forms for sharing their ideas.

Students’ reports of computational thinking practices

When reporting on their process of creating their e-textiles, students were required to describe either the design revisions they made or the problems they encountered along the way. While the requirements for these sections were simple (i.e., including problems and solutions, or revisions and justifications), some students’ more detailed descriptions referenced a number of rich computational practices. Below, we explore the computational practices that students chose to report on, including: 1) debugging and problem solving, 2) iteration and revision, 3) collaborating, and 4) using external resources and planning. Here we are less interested in the predominance of students who reported these computational practices (though we detail the percentages) than the actual presence of these descriptions in student-created portfolios, as they show the potential for process-based portfolios to document these under-assessed computational practices in students’ open-ended project design.

Debugging and problem solving

Within the Process section of portfolio, most students chose to explain at least one problem that arose while creating their e-textile designs (contained within 84.6% of all portfolios) (see Table 4 for subcategories). Almost all students who reported on a problem were able to successfully describe how they first observed this issue (83.7% of all portfolios), while a majority were also able to describe the specific solution they implemented (69.2%). While describing both a problem and its matching solution may seem like simple tasks, being able to identify and describe a particular issue within the context of a larger, complicated project is both a challenge and a recognized aspect of computational practices such as analyzing, decomposing, and debugging problems (e.g., Brennan & Resnick, 2012; College Board, 2017; Grover & Pea, 2018). The range of problems described in the portfolios were situated within all the

<table>
<thead>
<tr>
<th>Problems Discussed</th>
<th>Described (not mutually exclusive)</th>
<th>All portfolios</th>
<th>By students with experienced teachers</th>
<th>By students with inexperienced teachers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Observe an Issue/Problems</td>
<td>83.7%</td>
<td>93.8%</td>
<td>79.6%</td>
<td></td>
</tr>
<tr>
<td>Describe a Solution</td>
<td>69.2%</td>
<td>82.8%</td>
<td>63.7%</td>
<td></td>
</tr>
<tr>
<td>Test and/or Refine Something</td>
<td>19.5%</td>
<td>31.3%</td>
<td>14.7%</td>
<td></td>
</tr>
<tr>
<td>Iteratively</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Isolate a Problem/Root Cause</td>
<td>9.5%</td>
<td>15.6%</td>
<td>7.0%</td>
<td></td>
</tr>
<tr>
<td>Other</td>
<td>1.8%</td>
<td>3.1%</td>
<td>1.3%</td>
<td></td>
</tr>
<tr>
<td>At least one practice</td>
<td>84.6%</td>
<td>95.3%</td>
<td>80.25%</td>
<td></td>
</tr>
</tbody>
</table>
domains of e-textiles (coding, crafting, and circuitry), and often included overlaps in those domains, something that frequently occurs during the e-textiles construction process (Kafai et al., 2014).

To understand the type of thinking that went into documenting these problems, consider Valentina’s account of dealing with syntax problems which included an explanation of how these errors negatively impacted the functionality of her program:

One challenge in the coding was forgetting brackets and semicolons. First, when we initially tested the code, one light blinked and nothing else happened. While troubleshooting, we discovered that a few brackets were in the wrong place (causing the delays to loop) and a few semicolons were absent. To fix the issue, we checked each line for brackets and semicolons and our code carried out the desired function and was successful! (p 4)

In this excerpt, we see how Valentina is able to reflect on her process of debugging by breaking it down into multiple steps. This includes: 1) discovering the issue (“one light blinked and nothing else happened”), 2) diagnosing the cause (“a few brackets were in the wrong place … and a few semicolons were absent”), and 3) implementing a solution (systematically “checked each line for brackets and semicolons”). Further, we can see evidence of her strong comprehension of the code, where she successfully identifies how syntax errors can change the logical output of her program (i.e., how misplaced brackets were “causing the delays to loop”).

In the course of reporting on problem solving and debugging, a number of students also demonstrated engagement in the computational practice of isolating a problem or root cause from multiple possibilities (identified in 9.5% of all portfolios). This can be seen in the portfolio of Luz, who wrote about how she was able to diagnose the cause of her dysfunctional LEDs: “I hooked the [microcontroller] board up to a separate LED using alligator clips and it worked, so I knew that the issue was with the wiring on the inside” (p. 3). Here, Luz describes how she first had to check the functionality of her program by “hook[ing] up her board to a separate LED”, thereby isolating her written code from her sewn circuitry. After realizing that the code was indeed functioning properly, she then was able to identify that “the issue was with the wiring on the inside.” Finally, she was able to fix the stitched wiring so that the lights worked. Thus, even with this short quote, we can see Luz’ complicated process of moving across the three domains of e-textiles: from programming, to circuitry, to crafting.

Another debugging/computational practice that students were able to illustrate in their portfolios was testing and/or refining something iteratively (19.5% of all portfolios). Nino, for instance, (see Figure 3 above), wrote about the process of repeatedly adjusting the sensor reading ranges within his code to accommodate a range of users who would be touching the aluminum foil sensors in his interactive artifact:

When looking at the [sensor] values being received in Codebender’s serial monitor, I noticed that many of them were not included of the sensor value ranges required to activate a pattern … From testing with a group of friends, this seemed fine at first. However, when I gave the project to people that had more strength than us, the values of received went far less than 600 (The more pressure = lower value).

In order to make the project more accessible for anybody that wants to play around with it, I made a few changes within my code. I removed the minimum of 600 for the hard touch light pattern … [which] allows for anybody who is far stronger … to still
successfully turn on the pattern when applying their hardest touch … I also removed the maximum limit on the light pattern requiring the lowest amount of pressure … (p. 4)

Here, Nino demonstrates his ability to connect the multiple rounds of testing his project (both with “a group of friends” and “people that had more strength”) to the actual changes that he made to his code (changing the minimum and maximum pressure sensor readings written into the logical expressions of his conditional statements). He explicitly mentions how these changes are meant to make his “project more accessible” to a wider range of users. From this perspective, students’ communication of their processes make this practice of testing and refinement more visible to teachers and researchers, something that would normally be hidden when only evaluating a students’ final products.

Finally, a few students mentioned other miscellaneous problem solving practices (1.8% of all portfolios), including collaborative checking of problems, or learning from other people’s mistakes and solutions. Overall, students in classes with experienced e-textile teachers tended to show increased percentages of describing solutions, isolating problems, iterative testing/refinement, or other problem solving practices (see Table 4). As highlighted in our previous work (Lui et al., 2018), one major reason that might explain this difference is that these more experienced teachers had previous experience implementing the portfolios within the e-textiles curriculum. Based on this, they made changes in their classrooms to more explicitly promote high levels of computational communication in their classroom, leaving more time at the end of the unit for students to focus on creating their portfolios, as well as giving them greater opportunities to reflect and record their progress using the suggested journal prompts. Beyond these differences, however, students’ high levels of discussion of their debugging practice, as well as some students’ description of more complex computational practices such as isolating problems and testing/iteration illustrates the affordance of portfolios in not only sharing these experiences with their teachers, but also allowing students to externalize their thinking in the concrete format of the portfolio.

**Iteration and revision**

Students’ descriptions of their project revisions (40.3% of all portfolios) created additional opportunities to examine their engagement with computational practices. In our coding scheme, we categorized problem solving as the act of fixing some error during the process of construction, while revisions were categorized as changing the underlying design of a project for a wide range of reasons that did not necessarily relate to functionality. The most widely cited reason for changing a design related to spatial composition (22.2% of all portfolios), or the arrangement of components and/or circuit lines on the physical surface of the project. Within the portfolios, many students mentioned how spatial composition became difficult when dealing with the threedimensionality of projects. One student, Seth, demonstrated this when we described adjusting his design for his “stuffed taco” project and dealing with both inside and outside surfaces of the project. While he sewed his microcontroller to the inside of his toy, he did not realize that this would cause issues when connecting an external battery to the project. Changes he described included adding a little hole into his fabric to make plugging in the battery easier, as well as adding more “fluff” to the inside his project in order to isolate the stitches and prevent potential short circuits.
Table 5. Computational practices students described in revision-based process moments.

<table>
<thead>
<tr>
<th>Reasons for revision (not mutually exclusive)</th>
<th>All portfolios</th>
<th>By students with experienced teachers</th>
<th>By students with inexperienced teachers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spatial</td>
<td>22.2%</td>
<td>34.4%</td>
<td>17.2%</td>
</tr>
<tr>
<td>Aesthetics/Expression</td>
<td>11.8%</td>
<td>20.3%</td>
<td>8.3%</td>
</tr>
<tr>
<td>Usability/Audience</td>
<td>5.0%</td>
<td>14.1%</td>
<td>1.3%</td>
</tr>
<tr>
<td>Other Reason</td>
<td>7.2%</td>
<td>4.7%</td>
<td>8.3%</td>
</tr>
<tr>
<td>Reason not provided</td>
<td>4.5%</td>
<td>1.6%</td>
<td>5.7%</td>
</tr>
<tr>
<td>At least one reason</td>
<td>40.3%</td>
<td>57.8%</td>
<td>33.1%</td>
</tr>
</tbody>
</table>
surprisingly, many students in this class reported on these tests and subsequent revisions of their human sensor reading ranges in an attempt to accommodate different users.

In sum, all the revisions and related justifications described above illustrated the students’ iterative improvements on their projects, a key computational thinking practice. Additionally, discussion of these revisions also illustrates students’ engagement with other important, yet under assessed CT practices, whether creative expression through aesthetic decision making, attentiveness to usability and audience, or thinking about efficiency of computational design and work.

Collaborative work
Our open-coding also revealed students’ engagement with the computational practice of collaboration (see Table 6). Over one-third of the portfolios included descriptions of working with other people (35.8% of all portfolios), whether teachers, classmates, friends or family. Most students reported collaborations with their classmates (26.7%), with fewer writing about teachers (9.1%), and even fewer about unspecified or other people such as family (3.6%). As described in the sections prior, this can be seen within reports of working with peers on testing projects. Additionally, numerous students also mentioned actively receiving help from others, as Lili described: “I asked for help with my code from my fellow peer and teacher multiple times.” (p. 4). Reporting on these activities is notable since we did not explicitly ask for students to comment upon collaboration, and the portfolios were predominantly an individually graded project. Additionally, while CS is traditionally spoken about in individual terms (Yardi & Bruckman, 2007), this illustrates how e-textiles production can become a social practice, especially when considering the highly visible and tangible nature of the projects themselves (see Fields, Jayathirtha et al., 2019).

Planning and using external resources
Students also wrote about drawing from the wider environment while creating their e-textiles projects through use of external (non-human) resources, materials, and tools (8.6% of all portfolios). For instance, Serena wrote about referring “back on past [e-textiles] projects to help me find solutions” (p. 5), while Alicia described remixing existing sample code for fading LEDs, which she found in the Arduino IDE, for her own project. These reports therefore illustrated how students’ individual engagement in creation were still situated into the larger context of the unit and classroom (e.g., referencing past e-textiles projects) and in the broader DIY and open source communities of e-textiles (e.g., referencing sample code from the Arduino IDE).

<table>
<thead>
<tr>
<th>Collaboration</th>
<th>(not mutually)</th>
<th>All portfolios</th>
<th>By students with experienced teachers</th>
<th>By students with inexperienced teachers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classmate</td>
<td>26.7%</td>
<td>28.1%</td>
<td>26.1%</td>
<td></td>
</tr>
<tr>
<td>Teacher</td>
<td>9.1%</td>
<td>6.3%</td>
<td>10.2%</td>
<td></td>
</tr>
<tr>
<td>Other (family; neighbor)</td>
<td>3.6%</td>
<td>6.3%</td>
<td>2.6%</td>
<td></td>
</tr>
<tr>
<td>At least one collaborator</td>
<td>35.8%</td>
<td>37.5%</td>
<td>35.0%</td>
<td></td>
</tr>
</tbody>
</table>

Table 6. Types of collaborators students described.
Another practice that students mentioned within their process descriptions, but which was not explicitly asked for, was planning (27.6% of all portfolios). Discussions of planning included students’ explicit attempts to avoid problems and mistakes in the future, as well as organizing workflow for project completion. This can be seen within an excerpt from Dana’s portfolio, where she described her plan for sewing:

Sewing was pretty hard. I had to stitch from the back of the shirt to the front. Since the LED lights were in the back I had to make sure I had enough [thread] to make it around. The [thread] would sometimes get mixed up or get tangled. (p. 3)

Here, Dana’s description of her workflow (“stitch from back … to the front”), and its relation to her materials (ensuring she “had enough [thread] to make it around”) and potential issues to monitor (“mixed up or … tangled” thread) illustrates her thoughtfulness when thinking about issues of construction.

Relatedly, a number of students also mentioned the process of idea generation (8.6% of all portfolios) as part of their computational processes. Often mentioned as a problem to be overcome, students mentioned their “solutions” including drawing and sketching, looking through different objects or accessories at home, or talking through project ideas with others. As seen in Table 7, students with experienced e-textile teachers tended to report on planning and idea generation more frequently, again following trends throughout portfolio descriptions of more effective communication strategies, as well as sharing more information about their project-creation experiences.

Discussion

In this paper, we examined the assessment of the often ignored practice of communicating about computation. Just as students need to learn computational concepts and practices required for creating and debugging programs, they also need to be able to communicate about their products and processes. This is important not only for working with teachers and classmates, but any future collaborators down the line. Communication about computation can take various forms, from casual conversations with peers about troubleshooting their projects, to more formal documentation of code presented to users and clients. In our case, we chose portfolios as a multi-modal format to capture students’ communication about their e-textile products and processes.

Our findings reveal that students who created process-based portfolios documenting their e-textiles projects generally illustrated a solid ability to write and visualize computational ideas within their portfolios, albeit with room for improvement in several areas. The portfolios also allowed for students to report under-assessed computational practices.

<table>
<thead>
<tr>
<th>Table 7. Other computational practices described in Portfolios.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Other Computational Practices</td>
</tr>
<tr>
<td>Categories (not mutually exclusive)</td>
</tr>
<tr>
<td>Using External Resources (sample code; alligator clips; class notes; previous projects, etc.)</td>
</tr>
<tr>
<td>Planning</td>
</tr>
<tr>
<td>Coming up with Ideas</td>
</tr>
<tr>
<td>At least one other practice</td>
</tr>
</tbody>
</table>
such as debugging, iterating, and collaborating. Notably, the quality of students’ communication about their projects and processes in classrooms with experienced e-textile teachers (one or more years of e-textiles) was substantially better than those with novice e-textile teachers (teaching e-textiles for the first time). This difference in performance indicates how important both teacher preparation and classroom implementation are in supporting these communication skills. In the following sections, we (1) discuss what we learned about students’ abilities to communicate about computation, and (2) address challenges of implementing portfolios in classrooms.

Assessing computational communication

Our study shows that portfolios using both text and visuals can be used as an insightful form of assessment, especially for open-ended projects (not necessarily just e-textiles). Not only does the creation of a portfolio give students opportunities to rehearse this important skill, but also allows researchers and practitioners to capture aspects of their computational practice not easily captured by other assessments, such as their approaches toward debugging or their process of revisions. This knowledge of student learning would be difficult to gather through other means outside of time-consuming reflective interviews or think aloud problem solving sessions (e.g., Grover, 2017; Jayathirtha et al., 2020). For this reason, portfolios should be seen as a viable part of a larger “system of assessments” (Grover, 2017) of students’ computational thinking. In the e-textile unit, this system of assessments included small tasks such as design notebooks, journal entries, and short debugging tasks, as well as larger tasks, namely the set of four projects (each with a rubric for grading) and the portfolios.

In terms of the students’ communication ability, our rubric analysis shows that students who submitted portfolios possessed a solid aptitude in communicating about their computational projects and processes. As described in our literature review, students’ abilities to communicate about computation can be measured through three levels of competence (Taffe, 1989). The first level of competence describes students’ abilities to comfortably use the terminology/vocabulary of computing, while the second level describes their ability to apply this vocabulary to describing larger computational systems or projects. In our study, students demonstrated mastery of these first and second levels of communication, evidenced by the relatively high scores within the Product section of the portfolios, where students described the components and functionality of their own e-textiles projects. Here, students illustrated both their ability to communicate in written text and through visualizations, where we saw students using a variety of visual annotations (e.g., arrows; highlighted sections), as well as image formats (e.g., overall shots; close-ups) to support the description of their projects.

However, students still struggled with the third level of communicative competence (Taffe, 1989) which looks at how well students can marshall their computational descriptions to either support an argument, or provide evidence of a specific idea. In particular, we can see this in students’ varied abilities to describe their processes and reflect on what they learned (sections 2 and 3 of the portfolios, respectively). While some students were successful in detailing their experiences in these sections, others had more difficulty when attempting to write about more abstract ideas or procedures. For example, in the Reflection section of the portfolio, students tended to have an easier time providing
support for the hard skills they had learned (e.g., sewing, circuit design) in the form of pictures or specific details. However, discussion of soft skills learned (e.g., planning, design), though present, were more vague and lacking in evidence.

Another place where students’ struggled to communicate effectively was in describing their computational processes. Our findings highlighted the abilities of some students to describe less concrete methods such as testing and refining something iteratively (19.5% of all portfolios), and isolating problem causes during debugging (9.5% of all portfolios). However, not all students were able to describe their processes at this level of detail, nor include relevant images (whether in-process pictures or relevant code excerpts) to support this discussion. While we did provide supports and scaffolds such as in-class journal prompts and an exemplar portfolios, this finding supports the idea that even more scaffolds to aid students in capturing and accurately describing these moments are necessary. This is something we further discuss in the section on implementation.

Despite the issues with describing process though, our analysis did uncover aspects of CT that are rarely considered in lists of computational practices, such as considerations of audience and aesthetics (e.g., Brennan & Resnick, 2012; College Board, 2017; Grover & Pea, 2018). While under-discussed, these practices are legitimate areas of program design in making computational products more cohesive or responsive. This points to the need to expand our discussion of computational thinking to include social and cultural elements, as has been called for in computational participation (Kafai & Burke, 2014). Although collaboration on a product is a recognized computational practice, creating a product to be responsive or attractive to a specific audience is a far less recognized yet valuable area of professional computing.

Drawing from these findings then, one important consideration is the design of the portfolio assignment itself. This iteration of the portfolio provided specific constraints on students in terms of how they were allowed to report on their computational ideas and processes (i.e., requiring visualizations for each section, limiting process discussion to two problems or revisions). Considering the diversity of what students reported on, however, future iterations of a computational portfolio might provide more flexibility with regard to how students decide to communicate about their own projects, processes and learning. This not only includes allowing students to decide on the scope of their own descriptions, but also what kinds of media count as evidence. For example, while we required students to submit static visualizations for each section (e.g., pictures, images, code excerpts), our analyses demonstrated how these often were not the best way of providing evidence of abstract ideas or learning. This finding gets to the larger question of not only what counts as evidence when looking to assess students’ engagement with computational practices, but also considerations of how to best support students in making this determination on their own. While we focused on how students communicated about computation using text and visuals, future research might consider promoting communication through computation, such as creating a game, infographic, or animation to represent their experiences. Though beyond the scope of this study, providing flexibility within portfolios might be an avenue to promote greater agency for students, not only in terms of how they share their process with others, but also in shaping their own trajectories of computational growth and learning.
Implementing portfolio assessments in classrooms

Our findings also highlight another important item of consideration – how portfolios are actually implemented in Computer Science classrooms, and how this shapes students’ abilities to communicate about computation. This relationship is clearly demonstrated through the difference in performance between students with experienced e-textiles teachers versus novice e-textiles teachers. Not only did the former group perform better throughout areas of textual and visual communication, they also tended to provide much greater details about their processes of learning, therefore shedding light on commonly under-assessed computational practices. In addition, these experienced teachers tended to have greater portfolio completion rates than novice e-textiles teachers.

As described in the methods, one major contributor to this difference in performance and portfolio completion was the scheduling and timing of the unit. Teaching e-textiles is generally a time intensive activity because it not only incorporates new programming platforms (Arduino), but also adds logistical difficulties such as dealing with and storing tangible materials and tools (e.g., sewing needles, felt, conductive thread, scissors). As detailed in our prior research, our experienced e-textiles teachers took time to re-organize their space and re-orient their teaching practice to accommodate e-textiles creation (Fields, Lui et al., 2019), something which made their third year of teaching e-textiles more efficient and productive. Almost every student from these experienced classrooms finished all four suggested projects and also completed their portfolios on time. Novice e-textiles teachers, on the other hand, struggled with timing, with some only finishing three of the four suggested projects and prioritizing finishing a functional project rather than working on the portfolio. This points to the importance of considering scheduling when trying to add portfolios to existing computing curricula. However, because experienced e-textiles teachers were able to successfully incorporate portfolios into their teaching, we believe that this challenge can be easily managed with careful planning and scheduling.

Another major contributor to the difference in performance between experienced and novice teachers’ classrooms were how well they incorporated the portfolio throughout the course of the unit. While some teachers actively used the in-class supports we provided (e.g., journal prompts, exemplar portfolios), others did not. This was more prominent for experienced e-textiles teachers, who additionally made other adjustments in how they implemented the portfolios. For example, one experienced teacher ended up combining the portfolio assignment with a web design activity, such that students ended up spending more time on their portfolios than in other classrooms. Predictably, portfolios from experienced teachers’ students tended to contain greater levels of detail and visual evidence.

These differences in classroom implementation and their correlation with students’ communicative performance therefore suggests the importance of looking more closely at these factors in future research. Such studies need to capture not just the final days of creating portfolios, but teacher practices throughout the unit, for instance, encouraging students to pause during problems or mistakes that came up in their projects in order to both document and reflect upon their process. Another thing to note here was that this was the first process-based portfolio most students had ever created, so we might see this as a starting point for student learning rather than an endpoint. An additional line of research might consider how portfolio creation incorporated throughout the course of
a unit or a school year would help students develop better communication and reflection skills. From this perspective, what is needed are observations and documentation of best practices for facilitating computational communication of design processes and problem solving with e-textiles, and in other areas of open-ended computational design.

**Conclusions**

We examined high school students’ use of process-oriented portfolios to practice communication about computation. Overall, we found that students were capable of both writing about and visually documenting computational ideas, especially in the context of providing descriptions of their own computational designs. To a lesser degree, students were also able to report on their computational processes, demonstrating engagement with practices such as testing and/or refining something iteratively, and isolating problems and root causes. Because these portfolios were able to capture students’ engagement with these under-assessed CT practices, we consider these tools a powerful way to evaluate students’ computational performance, in addition to more traditional measures such as project evaluations and short-term programming tasks. That said, productive use of portfolios as an assessment tool depends not only on students’ individual engagement with this task, but also teachers’ ability to actively incorporate this activity into the classroom. This is supported by our findings that teachers’ prior experiences and current adaptations to support portfolio creation impacted the quality of students’ communication about computation both in terms of how detailed and clear the documentation was, as well as how many CT practices were described. In considering portfolios then, practitioners and researchers of CS education should additionally consider how to appropriately prepare teachers to create in-class scaffolds and ongoing opportunities for students to actively practice communication, especially in developing visualizations and reports on abstract computational processes or ideas. Only by considering all these factors can the true potential of computational portfolios for supporting communication and assessing CT practices be realized.

**Notes**

1. Hereafter, we refer to these three teachers as “experienced e-textile teachers” or “experienced teachers.” However, we want to emphasize that all teachers in the study were experienced CS teachers; it is simply the difference in prior experience with the e-textile unit that we study in regard to student performance on portfolios.
2. All text from student portfolios consist of direct quotes, with no corrections of grammar, spelling or punctuation.

**Acknowledgments**

This work was supported by a National Science Foundation grant to Yasmin Kafai [#1509245]. Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of either NSF, the University of Pennsylvania, Utah State University, or University of Texas, El Paso.
Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work was supported by the National Science Foundation [1509245].

Notes on contributors

Deborah Fields is an associate research professor of instructional technology and learning sciences at Utah State University. In her research she seeks to inspire and advocate for children’s creative expression with digital media, coding, and everyday craft materials. She works to break down stereotypes regarding who can create with digital media and computing. This includes projects which create educational opportunities in computer science to design with sewable electronics or the popular programming environment, Scratch. This interest carries over into the growing phenomenon of child-generated digital content in online environments. Her work has appeared in journals such as Mind, Culture and Activity, the International Journal of Computer Supported Collaborative Learning, and the Harvard Educational Review. She is a fellow of the International Society of Design and Development in Education.

Debora Lui is a high school engineering and computer science teacher at The Baldwin School. She earned a B.S. in Architecture and Business Management at MIT where she also holds an M.S. in Media Studies. Most recently, Deb completed a joint Ph.D. in Education and Communication from the University of Pennsylvania. She has taught at both MIT and University of Pennsylvania and designed curriculum for high school Computer Science classrooms. Her research focuses on the social and cultural contexts of students’ learning in maker education and computer science. Her work has appeared in journals such as Interactive Learning Environments and Transactions of Computer Education.

Yasmin Kafai is professor of learning sciences at the University of Pennsylvania. She is a researcher and developer of tools, communities, and materials for the promotion of computational participation, crafting, and creativity across K-16. She recently authored Connected Code: Why Children Need to Learn Programming (2014), Connected Gaming: What Making videogames Can Teach Us About Learning and Literacy (both with Quinn Burke), and Connected Play: Tweens in a Virtual Worlds (2013; with Deborah Fields) and co-edited Textile Messages: Dispatches from the World of Electronic Textiles and Education (2013; with Leah Buechley, Kylie A. Peppler, and Michael Eisenberg) and Beyond Barbie and Mortal Kombat: New Perspectives on Gender and Gaming (2008; with Carrie Heeter, Jill Denner, and Jennifer Y. Sun). Kafai earned a doctorate in education from Harvard University while working with Seymour Papert at the MIT Media Lab. She is an elected fellow of the American Educational Research Association and of the International Society for the Learning Sciences.

Gayithri Jayathiratha is a doctoral student in the Teaching, Learning, and Leadership division of the Graduate School of Education at the University of Pennsylvania. Her current research focuses on notional machines, debugging, and teacher practice in computer science education, project-based learning, and e-textile applications in K-12 classrooms.

Justice Walker is an assistant professor at the University of Texas, El Paso. His research in science and computer science education develops ways to encourage learning conditions that are both accessible and equitable. His educational work crosses domains of synthetic biology, ethics, design biology, and maker education. His work has been published in Teachers College Record and a number of ACM proceedings.

Mia Shaw is a doctoral student in the Teaching, Learning, and Leadership division of the Graduate School of Education at the University of Pennsylvania. Her current research focuses on identity
development, critical perspectives, and student voice in computer science education, project-based learning, and synthetic biology applications in K-12 classrooms.

ORCID

Deborah Fields http://orcid.org/0000-0003-1627-9512
Debora Liu http://orcid.org/0000-0003-3455-2949
Justice Walker http://orcid.org/0000-0002-4356-0396

References


Býrgýn, O., & Adnan, B. A. (2007). The use of portfolio to assess students’ performance. *Journal of Turkish Science Education, 4*(2), 75. https://d1wqtxts1xzle7.cloudfront.net/1988152/4gf0aho874un5c6.pdf?1425082536=&response-content-disposition=inline%3B+filename%3DThe_Use_of_Portfolio_to_Assess_Students.pdf&Expires=1609450788&Signature=EjWQqjt8VDVaW5ZxZbvbdGMoewfHj2-DrvCZYLalotysvbcKhXImXbbGQU2FiW1-i7UGA6-2t7b688PUP9jJ0IYWhZ9YuanP2X88-FfkYzw5A1dgUWWzJsIsh10EQKTfOpGqjbHZeiV425ZC6qL4LoxieCrzq3QtmGbbBAgNBNuvgiRPWunx8EXQkCnWSK4cvSRA-HTC6o3ozdq660shHU-sevagGgdcetEt4t9UZmwdW4HtB2LDZnAmzSPuo0uULm-BWYw8f1IMYVehODSwVbX05Vc1aqxyVFpAgYGpwclrwNWHLuRRFLxt38nlwskB2lsMUqjmqA-1gbetLZQsYQ__&Key-Pair-Id=APKAJLOHF5GSSLRBV4ZA


Jayathirtha, G., & Kafai, Y. B. (2019, February). Electronic textiles in computer science education: A synthesis of efforts to broaden participation, increase interest, and deepen learning. In
Proceedings of the 50th ACM technical symposium on computer science education (pp. 713–719). Minneapolis, MN.


