Learning Despite Resistance

Engaging Resistant Learners Through Creative Learning Activities

Ulf Dalvad Berthelsen (Danish School of Education, Aarhus University)

Abstract: Through a comparative analysis of two Scratch projects made by 2 different groups of 6th-grade students, this study shows how working with the visual programming language Scratch (www.scratch.mit-edu) provides resistant students with unexpected learning opportunities. The study compares the student projects with respect to level of code complexity, level of subject matter integration, and use of modalities and it is argued that the creative learning opportunities provided by the Scratch programming language encourage the resistant students to engage in meaningful learning activities despite their resistance.

Introduction

Resistant students are stereotypically thought of as disruptive troublemakers with low academic achievements. Resistance can have many different causes and take many different forms, but resistant learners all have one thing in common: Their resistance is assumed to prevent them from reaching the intended learning outcome (Alpert, 1991; Hauschildt & McMahon, 1996; Johannessen, 2003; Kim, 2010; Martinez, 2001; Reda, 2007; Williams, 2006).

The aim of this paper is to show that creative learning activities provide an excellent starting point for overcoming student resistance. The paper presents a comparative analysis of two digital student products created in the visual programming language Scratch. The two projects are created by a group of nonresistant students and a group of resistant students respectively, and the analysis points out important similarities and differences between the two projects in order to show how and to what extent the group of resistant students engage in the task. The paper concludes by discussing creative learning activities as a means to overcoming student resistance.

Background

The current study is part of a larger research project designed to explore how computer programming can be integrated across subjects in K–12 education. The two selected student projects result from a course in digital storytelling, in which 6th-grade students were asked to create digital stories in Scratch based on folktales such as *Snow White* and *Cinderella*. The course consisted of 10 weekly 90-minute sessions in which students worked in small groups collaborating on a digital story. For each session, the students first watched an instructional video explaining basic functionality in Scratch before moving on to develop characters, story line, and so forth in Scratch. Besides the instructional videos, the course material also included a booklet that helped the students organize the collaborative process and integrate subject matter into their Scratch projects.

The project resulted in several hundred Scratch projects varying from small test projects with only a few code blocks made by individual students to large projects with series of complex scripts made in collaboration between several students. Among these were many interesting and carefully worked out

projects suggesting that the participants had put a lot of effort into their animated stories. However, a handful of projects stood out. Not because they did not represent an effort, but because the effort was put into—in a very explicit and provocative manner—not complying with the task of animating a fairy tale. The two projects analyzed in this paper represent each of these two types of projects. The first is an animation of *Cinderella,* and the second is a project that shows no relation at all to the assigned fairy tale. The two groups of students attended the same class, and, hence, received the same instruction.

Methodology

From a user perspective, the visual programming language Scratch can be viewed as a multimodal typewriter. Instead of typing text, users drag and drop code blocks to create interactive animations, stories, and games by using a range of modalities such as text, color, sound, and two-dimensional motion. In order to capture this complexity, the two Scratch projects will be analyzed with respect to their formal structure in terms of code complexity and their content structure in terms of subject matter integration. Further, it will be discussed to what extent the learners use the modal affordances provided by the Scratch environment.

In order to compare the formal structure of the two projects, the analysis begins with a systematic review of the use of code blocks. There are six different block shapes (hat blocks, stack blocks, Boolean blocks, reporter blocks, wrap blocks, cap blocks) and 10 different categories (motion, looks, sound, pen, data, events, control, sensing, operators, more blocks), and each block, depending on its shape and category, represents a specific functionality in Scratch. A systematic overview of the use of code blocks, therefore, gives a good first estimation of the formal complexity of the projects.

The content of the two projects will be analyzed in terms of subject matter integration. The projects are the result of a creative collaborative process in which the students were asked to animate a fairy tale as part of a literature class. The course material supported the collaborative process by providing different kinds of scaffolds for developing characters, setting, and story line. Thus, an overview of how these elements were incorporated into the projects will give an estimate of the extent to which the students complied with the task.

Finally, the analysis focuses on the use of modalities. Cope and Kalantzis (2009) discuss a range of different modalities of which written language (on screen), oral language (recorded), visual representation (still or moving image), and audio representation (recorded music, ambient sounds, sound effects) are available in the Scratch environment. These modalities are fundamental features of contemporary communication, and an overview of how the students use these modalities will qualify the overall assessment of the student projects.

Code Complexity

Tables 1 and 2 give an overview of the formal structure of the two Scratch projects. Table 1 represents the project (NL) made by the nonresistant learners, and Table 2 represents the project (RL) made by the resistant learners. NL comprises nine sprites, six backdrops, and 80 scripts, and RL comprises 15 sprites, eight backdrops, and 41 scripts.

Sprites: 9 Backdrops: 6 Scripts: 80	Motion	Looks	Sound	Pen	Data	Events	Control	Sensing	Opera-tors	More Blocks
						xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxx (76)				
	xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xx (32)	xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxx (56)	xxxxxxxxxx xxxxx (15)			xxx (3)	xxxxxx (6)			
								xx (2)		
										
							xxxx (4)			
										

Table 1. Overview of project made by nonresistant learners.

Sprites: 15 Backdrops: 8 Scripts: 41	Motion	Looks	Sound	Pen	Data	Events	Control	Sensing	Opera-tors	More Blocks
						xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xx (42)				
	xxxxxxxxxx xx (12)	xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxx (53)				xxxxxx (6)	xxxxxxxxxx xxxxxxx (17)			
								xxx (3)		
										
							xxxxx (5)			
										

Table 2. Overview of project made by resistant learners.

Despite the difference in number of scripts (80 vs. 41), the graphic representation in Tables 1 and 2 reveals that the overall structures of the two projects are very similar. Except for the lack of sound blocks in RL, the students have made use of blocks from the exact same categories. The most conspicuous difference, which also relates to the difference in number of scripts, is the difference in the use of the event/hat blocks. As can be seen in Figures 1 and 2, the students who made NL chose a performative approach to creating a digital story in the sense that one has to "enact" the story by pressing a rather long sequence of keys in the correct order, whereas the students who made RL chose a more gamelike approach, in which one moves the sprites around in order to drive the story forward. As a result, NL contains almost twice as many event/hat blocks, mainly of the type "When [key] is pressed." This difference also explains the difference in the use of motion/stack blocks, since NL, because of the performative approach, contains a large number of individual motion events.

Equally interesting, both groups of students made only very limited use of code blocks representing

more complicated computational concepts. Except for event/hat blocks, which initiate events, the vast majority of code blocks in both projects consist of stack blocks representing simple events such as changing costume or moving the sprite around. In both projects, we find a few attempts to use control/wrap blocks for creating loops and sensing/Boolean blocks for creating simple interactions, but none of the projects uses data blocks or operator blocks for constructing complex functionality, and none of the projects uses control/wrap blocks for creating complex embedded structures.

Subject Matter Integration

Figures 1 and 2 show screen shots of the two projects. Figure 1 shows an inside view of NL, and Figure 2 shows an inside view of RL.

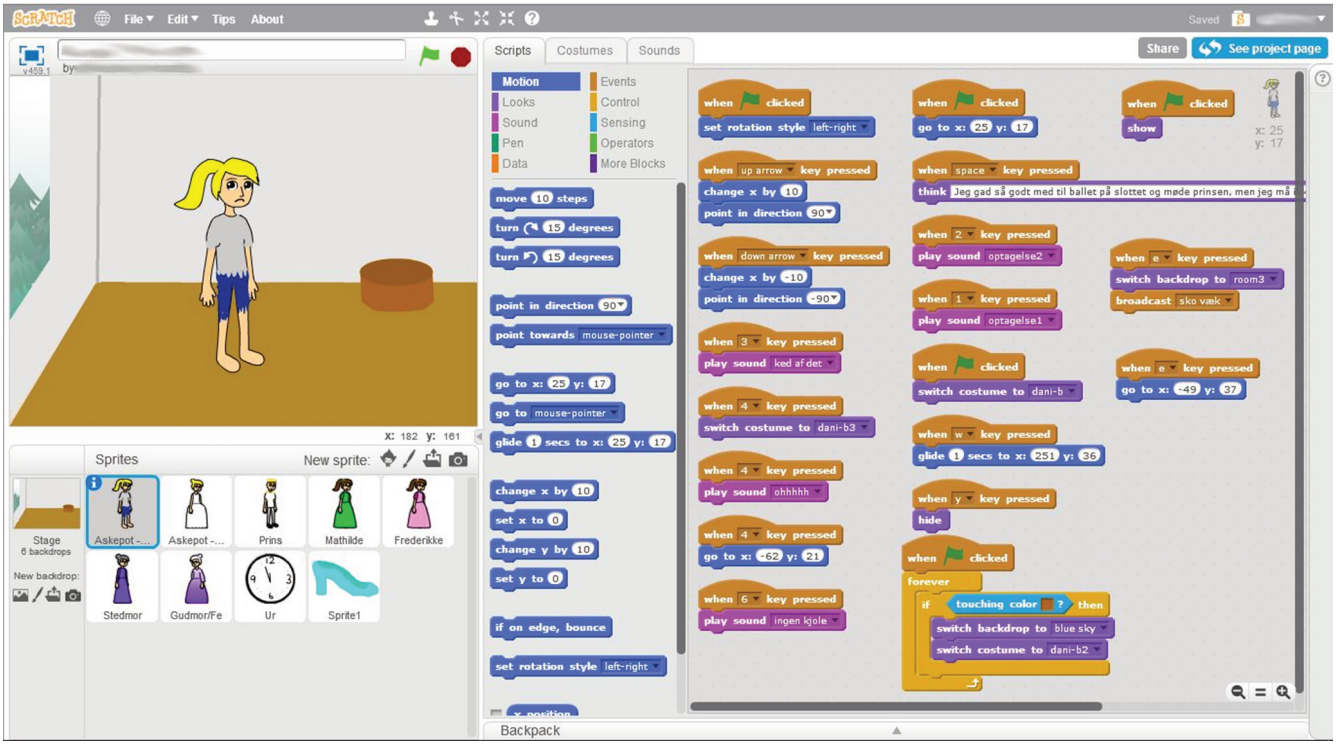

Figure 1. Screen shot of project made by nonresistant learners.

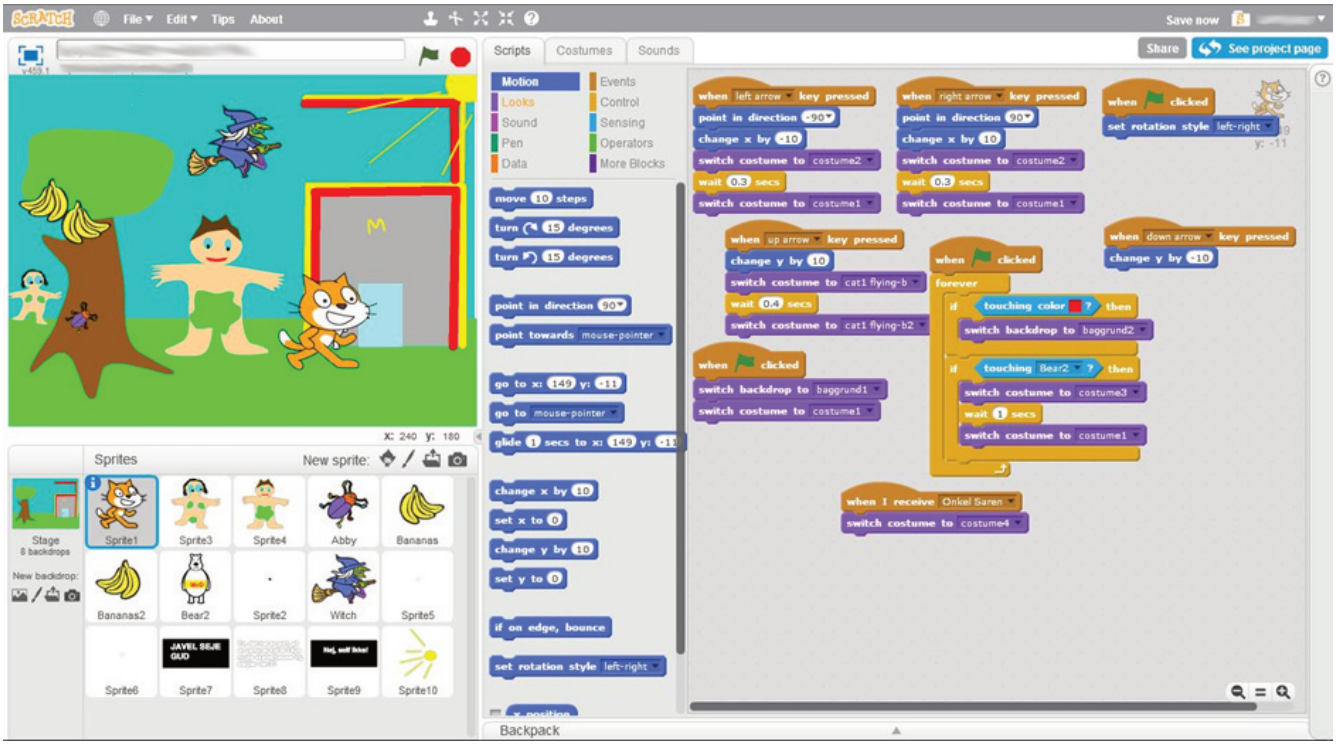

Figure 2. Screen shot of project made by resistant learners.

Despite their similar formal structure, the two projects differ significantly with respect to integration of subject matter. Most important, RL shows no relation at all to the assigned fairy tale. On the contrary, the project indicates that the students went to great lengths to create a project that demonstratively expresses their resistance toward the task. In contrast, the content and narrative structure of NL closely resembles that of the assigned fairy tale. The main differences with respect to subject matter integration, that is, setting, character development, and story line, are summed up in Table 3.

	NL (Table 1/Figure 1)	RL (Table 2/Figure 2)
Backdrops and setting	- Close relation to assigned fairy tale (Cinderella) - One backdrop per scene - Linear sequence of backdrops - Backdrops support storyline - Backdrops depict scenes from the assigned fairy tale (home, garden, castle)	- No relation to assigned fairy tale (Snow White) - Only one main backdrop - Events occur on the same backdrop - Unclear relation between backdrop and events - Unclear relation between elements in the project (McDonald restaurant, tree, landscape)
Sprites and characters	- Sprites represent characters from the assigned fairy tale - Clear internal relation between characters (Cinderella, the fairy, the step sisters, the prince) - All nine sprites are created by the students - Characters lines are taken from the assigned fairy tale - Proper costumes and language	- No relation between sprites and assigned fairy tale - Unclear internal relation between characters (a cat, a beetle, a witch, bananas, Adam and Eve) - Ten sprites created by the students, five sprites chosen from the Scratch library - No relation between characters lines and assigned fairy tale - Obscene costumes and offensive language
Interaction and storyline	- Users interact with the project by pressing keys in a specific order - Systematic relation between user interaction and story line - Linear narrative structure - Coherent story world - Recreation of the story world from the assigned fairy tale - Systematic relation between storyline and assigned fairy tale - Proper storyline	- Users interact with the project by pressing keys or clicking sprites with the mouse - No systematic relation between user interaction and storyline - Non-linear narrative structure - Non-coherent story world - Creation of own (story) world - Non-coherent story line - No relation between assigned fairy tale and storyline - Violent and provocative events

Table 3. Subject matter.

From the comparison above, it is clear that the students behind RL acted disruptive on purpose and deliberately failed to comply with instructions. They created an alternative story world comprising violent events, obscene pictures, and offensive language. For instance, there is no real time line, but when the key “j” is pressed, a sprite displaying text appears with a message, apparently from a god, ordering the other characters to kill the cat. When the “space” key is pressed, the character “Adam” loses his fig leaf and displays his genitals. When the beetle is clicked with the mouse, it is crushed, and when the “u” key is pressed, the entire setting blows up.

Surprisingly, this does not mean that there is no indication of integration of subject matter in RL. Obviously, we find a high level of subject matter integration in NL. The students have re-created scenes and characters from the assigned fairy tale using the drawing tools available in Scratch, and they have used the functionality of the code blocks to re-create the story line. This is not the case in RL. However, if we disregard the provocative and offensive nature of the project as well as the fact that the students behind RL have deliberately tried to disrupt the course, it becomes clear that we, after all, can find examples of subject matter integration in the project. For instance, the students have engaged in developing an appalling setting, they have put effort into making up violent scenarios, and they have put effort, as well as a certain amount of humor, into developing the Adam costumes.

Modality

As mentioned earlier, the Scratch environment can be seen as a multimodal typewriter providing a range of different expressive means, including written language, recorded oral language, visual representation,

and sound. When the two projects are compared with respect to these modalities, there appear to be both similarities and differences (see Table 4).

Modality	NL (Table 1/Figure 1)	RL (Table 2/Figure 2)
Text	- Looks/Stack block 'Think [text]' - Sprite containing numbers on a watch - Backdrop containing text 'The End'	- Several sprites containing text - Backdrop containing larger amount of text
Oral language	- Sound/Stack block 'Play sound [recorded speech]'	
Visual representation	- Sprites created as bitmap and vector graphics - Backdrops created as bitmap and vector graphics - Motion/Stack Block 'Change x by [value]' - Motion/Stack block 'Go to [x,y coordinates]'	- Sprites created as bitmap and vector graphics - Backdrops created as bitmap and vector graphics - Motion/Stack Block 'Change x by [value]' - Motion/Stack Block 'Change y by [value]' - Motion/Stack block 'Move [value] steps'
Sounds	- Sound/Stack block 'Play sound [sound effect from library]'	

Table 4. Modality.

The most important difference between NL and RL is that RL is made entirely without using the sound modality, neither recorded speech nor sound effects from the Scratch library. Concerning the other modalities, both projects use them in a variety of ways. Both projects integrate written language to some extent, and both projects make extended use of visual representation.

Disruptive but Creative

The comparison between NL and RL has shown that the two projects have a lot in common with regard to code complexity and also, to some extent, with regard to the use of modalities. However, there are also important differences, especially in relation to the content of the projects. The students behind NL complied with the task and followed instructions. The students behind RL, on the other hand, put a lot of effort into not complying with the assigned task and into creating a provocative project containing both obscene pictures and offensive language.

Thus, the analysis concluded, one important thing stands out. The resistant students showed both effort and engagement. Consequently, and despite their resistance to learning and learning activities, they were presented with a range of unexpected learning opportunities. Brennan and Resnick (2012) argue that

the Scratch environment provides learners with the opportunity to engage with important computational concepts. As the analysis showed, the resistant students were also presented with the opportunity to engage with multimodal means of expressions and, to some extent, also with literary concepts.

One might think that the students' resistance and disruptive behavior prevented them from learning, but here it is important to notice that they actually engaged in creating a multimodal digital product when they could have chosen to do nothing at all. This suggests that creative learning activities, in this case working with Scratch, possess qualities that encourage resistant learners to engage in learning activities despite their resistance.

Of course, the project was also inacceptable in many respects. The obscenity and offensive language have already been mentioned, but the project is also disruptive in a way that disrespects both the teacher and the peer students in the class. Thus, the point to be taken is not that this behavior should be accepted, but rather that the engagement of the resistant students provides an excellent starting point for the process of overcoming resistance.

Engaging Disruptive Learners Through Creative Learning Activities

It lies outside the scope of this paper to investigate what caused the unexpected engagement. However, following Papert (1980), it seems plausible to assume that the creative learning opportunities provided by the Scratch environment provide agency to the resistant students. Along these lines, Resnick (2017) suggest projects, passion, peers, and play as important features of creative learning activities. Features that, one hopes, will support resistant students on their path to becoming creative learners and provide teachers with means to scaffold the transformation.

References

Alpert, B. (1991). Students' resistance in the classroom. *Anthropology and Education Quarterly, 22*(4), 350–366.

Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. In *Annual American Educational Research Association meeting*, Vancouver, BC, Canada.

Cope, B., & Kalantzis, M. (2009). "Multiliteracies": New literacies, new learning. *Pedagogies: An International Journal, 4*(3), 164–195.

Hauschildt, P. M., & McMahon, S. I. (1996). Reconceptualizing "resistant" learners and rethinking instruction: Risking a trip to the swamp. *Language Arts, 73*(8), 576–586.

Johannessen, L. R. (2003). Achieving success for the "resistant" student. *The Clearing House, 77*(1), 6–13.

Kim, J. H. (2010). Understanding student resistance as a communicative act. *Ethnography and Education, 5*(3), 261–276.

Martinez, M. (2001). Key design considerations for personalized learning on the Web. *Educational Technology & Society, 4*(1), 26–40.

Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. New York, NY: Basic Books.

Reda, M. M. (2007). Re-seeing resistances: Telling stories. *Composition Studies, 35*(2), 31–52.

Resnick, M. (2017). *Lifelong kindergarten: Cultivating creativity through projects, passion, peers, and play*. Cambridge, MA: The MIT Press.

Williams, B. T. (2006). Metamorphosis hurts: Resistant students and myths of transformation. *Journal of Adolescent & Adult Literacy, 50*(2), 148–153.

Acknowledgments

The research reported in this paper was supported by an AUFF Starting Grant from Aarhus University Research Foundation.