Coding for Outdoor Play: A Coding Platform for Children to Invent and Enhance Outdoor Play Experiences

Netta Ofer
Media Innovation Lab (miLAB)
Interdisciplinary Center
Herzliya, Israel
netta.ofer@milab.idc.ac.il

Idan David
Media Innovation Lab (miLAB)
Interdisciplinary Center
Herzliya, Israel
idan.david@milab.idc.ac.il

Hadas Erel
Media Innovation Lab (miLAB)
Interdisciplinary Center
Herzliya, Israel
hadas.erel@milab.idc.ac.il

Oren Zuckerman
Media Innovation Lab (miLAB)
Interdisciplinary Center
Herzliya, Israel
orenz@idc.ac.il

Abstract

Outdoor play is in decline, including its benefits to children's development. Coding, a typically indoor, screen-based activity, can potentially enrich outdoor play, serving as a rule-making medium. We present a coding platform that controls a programmable hardware device, enabling children to technologically-enhance their outdoor play experiences by inventing game ideas, coding them, and playing their games together with their friends. In the evaluation study, 24 children used the system to invent and play outdoor games. Results show children are able to bridge between the different domains of coding and outdoor play. They used the system to modify traditional games and invent new ones, enriching their outdoor experience. Children merged computational concepts with physical game elements, integrated physical outdoor properties as variables in their code, and were excited to see their code come to life. We conclude children can use coding to express their ideas by creating technologically-enhanced outdoor play experiences.

CHI 2019, May 4–9, 2019, Glasgow, Scotland Uk

ACM ISBN 978-1-4503-5970-2/19/05…$15.00
https://doi.org/10.1145/3290605.3300394

Figure 1: Children coding and playing their outdoor game (children photographed with permission).

CCS Concepts

• **Human-centered computing → User interface management systems**; • **Applied computing → Interactive learning environments**;

Keywords

Games/Play, Children, Programming, Outdoor play

ACM Reference Format:
Netta Ofer, Idan David, Hadas Erel, and Oren Zuckerman. 2019. Coding for Outdoor Play: A Coding Platform for Children to Invent and Enhance Outdoor Play Experiences. In *CHI Conference on Human Factors in Computing Systems Proceedings (CHI 2019), May 4–9, 2019, Glasgow, Scotland Uk.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3290605.3300394

1 Introduction

Compared with children in the 1970s, children today spend 50% less time in unstructured outdoor play [29]. This decline is attributed to several factors, including parents' safety concerns, lack of green spaces for play [30], and time spent using technology indoors [23]. Outdoor play is known to have positive effects on child development, including practicing problem-solving, creative thinking, and abstraction [1, 11, 16]. The open-ended, unstructured nature of outdoor activities [8, 16], provide opportunities to acquire social skills [5, 6, 10, 16] and fosters physical play [6, 28, 29, 39].

A key aspect of outdoor play, and the focus of this paper, is rule-making [2, 21, 27]. Rule-making is considered a basic feature of play as it provides structure to the play activity while preserving its open-ended nature. In outdoor play, children can define and change the rules however they desire [4, 8–10, 33], contributing to physical play and social interaction [9].

Several researchers have defined and classified rule-making in play. Caillois (2001) defined play as an activity that disregards rules from daily life and provides children with the opportunity to define their own intrinsic, specific set of rules, for the specific space and time of their play [14, 46]. Salen and Zimmerman (2004) suggest that play involves three types of rule: (1) Operational rules, that are the concrete and specific actions that players need to take in the game; (2) Constitutive rules, that define the abstract, logical, formal structure of the game; and (3) Implicit rules, that define the game ethics through "unwritten rules" [42]. Children typically modify games by changing the operational rules in creative ways, ultimately creating their own custom play experience [27].

Rule-making is also a core principle of coding. In coding, logic rules are used to establish relations between variables or objects, by using algorithmic thinking and computational thinking concepts such as conditions, sequential execution, and parallelism [17, 18]. When children engage in coding, they are required to generate rules and plan their execution order [18]. This is thought to develop algorithmic thinking and categorization skills [18, 22], known to enhance rule-based thinking in other domains such as rule finding tasks and problem-solving [22].

In this study, we explore if and how the rule-based nature of coding can serve as a new medium for children to create their own outdoor play experiences. By integrating coding with outdoor play, we hope to promote two goals: (1) leverage technology to enrich outdoor play; and (2) leverage outdoor play as a new way for abstract code to "come to life" and unfold into a physical and social experience.

Adding coding to outdoor play can leverage technology to enrich traditional play and create new play experiences. Technologically-enhanced outdoor play has been suggested as a means to address the decline in outdoor play [3, 33]. Coding as a rule-making medium is a unique approach, and may possibly enhance the range and scope of play experiences children invent.

Experience with coding from a young age is thought to extend children's ability to design, create, and invent [37, 40, 41]. Coding activities are considered to promote analytical thinking, abstraction abilities, [7, 40, 47, 48], and computational literacy [20]. As coding becomes common in many schools, a wide range of approaches should be developed for introducing coding to diverse audiences. Outdoor play adds a unique nature to coding, and may appeal to children in ways that traditional coding activities do not, especially to children who are "outdoor play-fluent" [19].

However, the integration of coding and outdoor play is not trivial. Outdoor play has known characteristics, namely social interaction and physical activity [11, 16], as well as "heads up" play patterns [31, 43]. Coding is typically a screen-based, "heads down" activity, and may compromise outdoor play characteristics. Careful consideration should be given to balance the advantages of coding with the disadvantages of adding technology to outdoor play [31].

We present a tablet-based coding platform and a programmable hardware device. Our device, previously published, was specifically designed for outdoor play [26]. Children with prior interest in coding may be motivated to use this system to enrich outdoor play for themselves and for their friends, whereas children who aren't naturally drawn to coding activities may be motivated to code when set in outdoor play context. We evaluate if and how children bridge the gap between coding and outdoor play and use code as a medium for creating technologically-enhanced outdoor play experiences. We further assess if children are able to merge code with the physical world in a meaningful way, and use the "heads down", screen-based coding activity to create "heads up" outdoor play activities, without compromising outdoor play characteristics.

2 Related work

The HCI community has suggested several approaches for enhancing outdoor play with technology, mainly in the Pervasive Games and Head Up Games (HUG) domains [3, 33, 44].

Pervasive Games

Pervasive games are designed to take gaming away from the computer screen and into the physical world, leveraging technology to merge the physical and digital game elements [31, 35]. Sensing in Pervasive games integrates natural obstacles and challenges from the physical world into the game, and digital feedback brings game elements into outdoor play. This results in games that require physical effort and problem-solving in a real-world context [31]. Two notable examples are Pirates! and PacMap. In Pirates!, handheld computers and real-world properties such as locations are used as elements of the game, encouraging players to complete missions by exploring their digital-physical surrounding [12]. In PacMap, a location-based variant of the classic PacMan, players need to avoid enemies and collect rewards in real streets [15].

Head Up Games

Pervasive games commonly involve screens and hand-held devices that may interfere with natural outdoor play patterns [34], and may compromise the benefits of play [44]. To address these concerns, Markopoplus & Soute (2007), defined a sub-category of pervasive games coined Head Up Games (HUG) [43]. HUGs strive to promote outdoor social interaction with the support of digital devices while keeping players heads "up" to encourage natural social interaction. For example, "Stop the Bomb" prototype [25] is a belt with vibration motor and LEDs. Children are assigned to teams based on LED color, and vibration feedback is used to indicate which player holds the key. HeartBeat prototype [32] is a small portable device with heart rate sensing as input and a buzzer that beeps if the rate of the opponent's heart beat exceeds a preset value.

More recent HUG prototypes include traditional, digitally-enhanced outdoor play objects (e.g. stick, ball), rather than adding new designs for play devices. Relevant commercial Kickstarter projects include Hackaball [24] and Storyball [45]). Our previous work on HUG includes a digitally-enhanced stick as a research platform for digital outdoor play [26]. We evaluated our design and showed that children's play patterns with the stick-like device were similar to traditional outdoor play patterns.

Rule-making platforms for outdoor play

Some studies in the HUG domain have also developed platforms allowing children to change simple parameters while playing interactive outdoor games. Avontuur et al. (2014) developed RaPIDO, a sensor-based prototype of a screen-less, hand-held device, and GameBaker, an accompanying platform for parameter-changing [2]. By changing a set of pre-defined parameters such as buzzing duration and the number of participating teams, children could create and modify outdoor games based on their own ideas. Their evaluation indicated that children were enthusiastic about using a digital platform to create and modify their own games [2].

In our previously published Work-in-Progress, we presented a preliminary rule-making platform for outdoor play [36], with four programming blocks (two Event blocks and two Action blocks) that control simple input and output commands for our outdoor play device. The pilot study revealed a trade-off between children's coding and game invention activities. Some children were interested in coding and ignored the outdoor play context, while others were interested

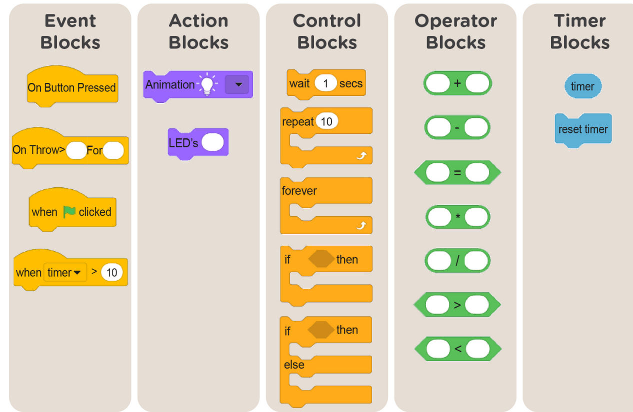

Figure 2: The coding platform's blocks include Event blocks, Action blocks, Control blocks, Operator blocks, and Timer blocks.

in inventing and playing outdoor games and used the code in a minimal way.

We extend this Work-in-Progress by implementing a comprehensive coding platform for programming the stick-like device, including events, conditions, parameters, operators, and control blocks. Using the platform, children can control the device's inputs, outputs, and logic. The coding platform can be used as a rule-making platform for defining, editing, and changing outdoor game rules. We evaluate the system with 24 children, assessing if and how they code outdoor play experiences.

3 Design and Implementation

We present the implementation of a block-based coding platform, designed to control the functionality of our previously-published hardware device designed specifically for outdoor play [26]. The device resembles a stick, enabling children to hold, throw, and catch it. It has a 3D printed inner case for holding the hardware, an outer case for protecting the hardware from intense physical force, and rubber caps to soften falls. The electronic components include a Feather M0 Arduino compatible board, an nRF51822 Bluetooth Low Energy (BLE) for communication (built in on the Feather M0), a BNO055 Inertial Measurement Unit (IMU), a push button for sensing, and 27 LEDs arranged in three columns to present feedback to the user.

Coding Platform

The coding platform consists of 20 custom drag-and-drop blocks, implemented using the Scratch 3.0 codebase, enabling children to use code primitives to program the device's inputs, outputs, and logic. We implemented five block categories: Event blocks, Action blocks (termed "Stack blocks" in Scratch), Control blocks, Timer blocks (termed "Sensing blocks" in Scratch), and Operator blocks (see Figure 2).

Following is a detailed list of all blocks. The "()" sign represents a placeholder for a parameter value to be entered by the user. In the blocks implementation, this placeholder is marked as an empty space for easier understanding by children. When the empty space is pressed, an input numeric dial-pad pops up.

Event blocks include the following blocks : 'When button pressed', 'When device in motion > () for ()', 'When "Green flag" pressed', and 'When timer > () seconds'.

Action blocks include the following blocks: 'Play animation', and 'Add () LEDs'.

Control blocks include the following blocks: 'Wait () seconds', 'Repeat () times', 'Forever', 'If (), then ()', and 'If (), then (), else ()'.

Timer blocks include the following blocks: 'Timer', and 'Reset timer'.

Operator blocks include the following blocks: '()+()', '()-()', '()=()', '()*()', '()/()', '()>()', and '()<()'.

Children can assemble code configurations (the blocks used for the code and the way they are connected as a program) with blocks from all five categories. The minimal example of a code configuration is an Event block followed by an Action block, for example: 'When button pressed', 'Play animation'. Children can construct various code configurations and as a result create many Code Rules. The device programming process has 4 steps: (1) Select blocks and create code configurations; (2) Press the "Green flag" button to send the code to the physical device; (3) The code is sent via BLE and is automatically executed on the hardware device; (4) Children experience the code they created through real-time interaction with the physical device.

To assess coding flexibility influences on children's coding, we created two different versions of the coding platform: version A involved a full coding experience, that includes both coding blocks and parameter manipulation within each block; version B had a more limited experience, with the exact variety of coding blocks but with no ability to manipulate parameters within each block. For example, the value of repetitions in a "repeat" block in version A will be editable by the child but will be fixed to a default of five repetitions in version B.

Platform & Firmware Implementation

The software implementation was optimized for a tablet device, based on Scratch 3.0 VM, including custom Scratch blocks and a Java-based transport layer between the blocks and the physical device. We chose to transfer as much computational load as possible from the physical devices to the tablet device, since memory and computing power are limited on the hardware devices.

The various software modules interact in the following way: the blocks activate the transport layer, which sends the events to the physical device via BLE. A listening service (firmware on the Arduino board) runs on the device's hardware with a set of predefined behaviors, such as acceleration threshold. The device receives the event parameters and when an event occurs, the appropriate blocks are activated by the transport Layer.

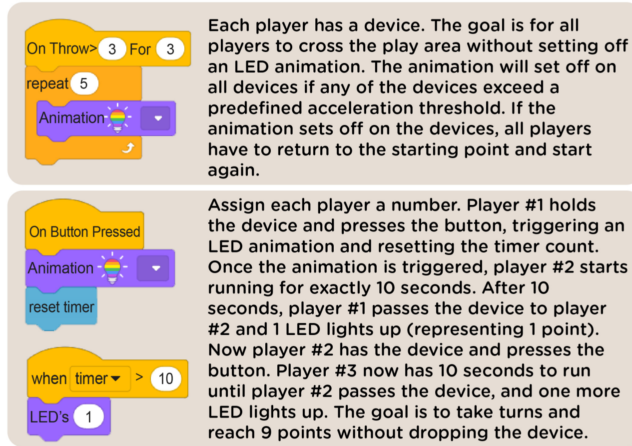

Figure 3: The code blocks and the Game Rules of the two introduction games presented to the children in the introduction stage.

A log system was implemented to enable easy tracing of the children's coding process, without interrupting the user's natural activity. Every time the user pressed the "Green flag" button, which sends the code from the tablet device to the physical device, two logging actions were launched: (1) a screenshot was automatically captured and stored on the tablet with the matching time stamp; (2) a full activity log was saved, enabling the replay of every code action the user performed since the last logging activity. This dual logging technique enables the researchers to trace all coding activities for later analysis. It also enables accurate recording of each iteration or debugging event the user performed, because an iteration must include an activation of the "Green flag" button by the user, to send the code to the physical device. During the data analysis, the researchers were able to observe and analyze the screenshots and activity log, replay the games created and notice every change (minor or major) performed in the code.

4 Evaluation

We evaluated children's ability to use the coding platform and programmable device in the outdoor play context. We tested whether they can use the system to enrich traditional outdoor play and create new game experiences. We evaluated children's ability to integrate the playful, social, and physical aspects of outdoor play with their abstract code. We further considered that the basis for every coding experience is the ability to form a sequence, use events, loops, and conditions. These processes are all supported by our platform's coding blocks. We were interested to learn if the flexibility of coding influences children's ability to code for outdoor play. We therefore used the two versions of the coding platform (regular coding experience, and a limited coding experience). Half of the participants received version A (regular) and the other half received version B (limited).

Method

The study took place at an outdoor play area and children were given the opportunity to use the system for creating new game experiences with the programmable device. To assess whether children can bridge the gap between coding and outdoor play, we evaluated the correspondence between the games children invented and the code configurations they created to control the device. Two terms were introduced to the children, Game Rules: the players' instructions, the goal, and objectives of the game; and Code Rules: the code used for programming the device for the game. Bridging the gap between coding and outdoor play was defined as whether children succeeded in programming the device to be compatible with the Game Rules they invented in a way that their overall game was coherent and playable. We further evaluated the types of games children invented, the role assigned to the device in their games, and the expression of their abstract code in outdoor play context.

Participants 24 children, divided to 8 groups of 3, with prior acquaintance within groups (8 girls and 16 boys; age range 9-12, Mean = 10.33; SD = 0.80; all children from a mid-high socioeconomic status), participated in the study. In order to ensure a unified level of programming experience, all children had a basic experience with Scratch. Children were recruited in groups of 3 with prior acquaintance from two sources: personal acquaintance with the researchers, and a local coding event ("Scratch Day"). Prior acquaintance of the children within each group ensured a natural social interaction between the children, a critical aspect for natural outdoor play. We followed ethics guidelines including Read's (2015) guidelines for research with children [38], IRB, parental consents, children consent, and parental approval for taking pictures and videos. Children who participated in the research received a guided tour of the research lab.

Procedure Children were invited to create their own outdoor games using the system, in a grass-covered outdoor play area of approximately 400 square meters with surrounding trees. The play area was marked by a ribbon to encourage exploration but to keep children in the researcher's field of view. All sessions were documented via a video camera and wireless microphones worn by the children. The activity began with an introduction of the programmable device. The researcher presented the device as a programmable outdoor play device with a motion sensor, a button, and LED lights. The researcher explained that the device can be programmed with a block-based coding platform, similar to Scratch. The activity consisted of three main stages: an introduction to the programmable device; introduction to the coding platform and game invention activity with the system; and a post-activity group interview followed by playing the invented games (see Figure 4).

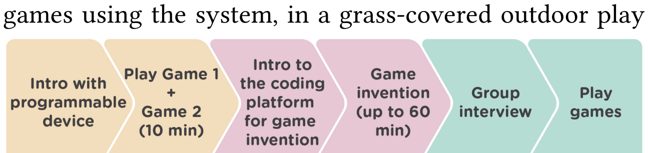

Figure 4: The experimental procedure.

Introduction games

We invented two game examples for the system introduction stage, using both the Game Rules (operational rules) and the code configurations (representing the constitutive rules). The first game was a modified version of a traditional outdoor game. The Game Rules were "Each player has a device. The goal is for all players to cross the play area without setting off an LED animation. The animation will set off on all devices if any of them exceed a predefined acceleration threshold. If the animation sets off on the devices, all players have to return to the starting point and start again". The Code Rules for this game were: 'When device in motion > 3 for 0.3 seconds', 'Repeat' 'Play animation' for 5 times.

The second game was a new game that included only one device. The Game Rules were: "Assign each player a number. Player #1 holds the device and presses the button, triggering an LED animation and resetting the timer count. Once the animation is triggered, player #2 starts running for exactly 10 seconds. After 10 seconds, player #1 passes the device to player #2 and 1 LED lights up (representing 1 point). Now player #2 has the device and presses the button. Player #3 now has 10 seconds to run until player #2 passes the device, and one more LED lights up. The goal is to take turns and reach 9 points without dropping the device". The Code Rules for this game were: 'On button pressed', 'Play animation', 'Reset timer'. 'When timer > 10 seconds', 'Add 1 LED' (see Figure 3).

Stage 1: Introduction to the programmable device The activity began with an introduction to the programmable devices by asking children to play the two example games (games were pre-programmed by the researcher). The purpose of presenting these games was to present the children with the different features of the system while preserving the outdoor play context. The children received Game Cards, which are paper cards with written text listing two type of

Table 1: Dependent measures.

Source	Measure	Description
Code log files	Number of iterations	Number of times the child "compiled" their code by clicking the "Green flag" button
	Total coding time	Total time spent coding
Game Cards	Game type	Modified version of an existing game Novel game Similar to one of the introduction games
	Role assigned to the programmable device	Game facilitator Game objective New game dimension
	Physical manifestation of code in outdoor play context	Game manifestation in the physical or social context of outdoor play
Code log files and Game Cards	Correspondence between the code configurations and the invented Game Rules	Full Correspondence or Partial Correspondence
Group interview	Enriching outdoor play	Comments regarding uses of the technology in an outdoor play context, and integration of technological aspects into outdoor play
	Physical manifestation of code in outdoor play context	Comments regarding the physical and social expression of their code

rules: Game Rules and Code Rules. The Game Rules define the game instructions and the Code Rules describe how the game was implemented in the code. Together with the Game Cards, each child received one programmable device, and the children were invited to play the game for 5 minutes. This process was repeated for each of the two games. During this introduction stage, the coding platform was used only by the researchers and not by the children. The two games represented different types of games, the first was a modified traditional game using 3 devices, and the second was a new game using one device (see Figure 3).

Stage 2: Introduction to the coding platform and game invention activity After a short water break, the children returned to the play area and were invited to create their own games. A researcher introduced the coding platform, informing them that the system is based on Scratch blocks, and has custom blocks such as 'When in motion', 'When button pressed', 'Play animation', and 'Set () LEDs'. Each child was given a printed "Block Index", listing all of the system's blocks and their function (a total of 20 blocks; see Figure 2). The researchers described some of the blocks' functions, and invited the children to explore and code with their own tablet and programmable device. In addition, each child received several blank Game Card templates to document their games.

With both the programmable device and the coding platform, children were free to select blocks and assemble code configurations based on their ideas, and then send their code configurations to the device to be played according to their Game Rules. Children could created code configurations and test the outcome on the programmable device. Throughout the activity, children could iterate between generating game ideas, coding, debugging, testing the game with friends, and documenting their game with the Game Card template. The groups were given up to 60 minutes for the game invention stage. As some children requested assistance from the researchers, a structured assistance protocol was defined. The protocol consisted of predefined answers to children's questions including:

- Think of games they know and like if no game ideas come to mind.
- Referring to the Block Index when looking for a specific block.
- Running a test run to find what isn't working in the code.
- Uploading the code to the programmable device.
- Resetting the system with the 'Stop button'.

Children who completed the activity before the activity ended were invited to further extend their game or create a new game.

Stage 3: Post-activity group interview and playing the invented games A short group interview was conducted when all children completed the activity in order to evaluate children's attitudes toward coding for outdoor play. The researcher asked the children to describe their experience with the system, the difficulties they had while using the system, and what they would change or add to the system. At the end of this stage, children were invited to play the games they created with their friends (see Figure 1).

Dependent Measures The dependent measures were based on data from the code log files, The Game Rules documented in the Game Cards, and group interviews. The dependent measures are listed in Table 1.

5 Analysis

The code log files were analyzed to extract exact "number of iterations" and "total coding time" for every child, and were averaged across children (see Table 1), representing a qualitative indication for children's engagement. In addition, the code configurations from the final code log file were reviewed and compared to the matching Game Rules taken from the Game Cards to assess whether the child successfully bridged the gap between outdoor play and coding. Two researchers examined all pairs and determined the level of correspondence between the children's coding configurations and their written Game Rules: Full Correspondence and Partial Correspondence. Full Correspondence means that after programming the device, (1) children are able to successfully play the game; and (2) all code configurations were meaningful to the Game Rules and represented all Game Rules that involve the programmable device. Partial Correspondence means that *not all* code is meaningful to the Game Rules, or some Game Rules are missing representation in the code. See Figure 5 for an example of a game from each level. A third researcher then coded half of the paired code log files and Game Rules, and interrater reliability was calculated (Kappa = 0.85). The few inconsistencies between raters were discussed and resolved. All measures and their sources are presented in Table 1.

The Game Cards were also used to identify the types of games children created and the role assigned to the programmable device in the game. These measures and the group interviews were analyzed using thematic coding [13] that included three stages. First, the primary rater reviewed all transcripts and identified initial emerging themes, presented them to a second researcher and discussed in depth, inconsistencies were resolved, and a list of mutually-agreed themes was defined. Second, the primary rater and an additional rater analyzed a selection of the data independently, interrater reliability was tested and found to be high (Kappa = 0.92). Third, following the interrater reliability validation, the two raters analyzed the rest of the data.

6 Findings

The qualitative and quantitative analysis revealed that children were able to bridge the gap between coding and outdoor play. Children were highly engaged and iterated between coding with the platform and testing their code with the device. They spent a long time thinking, planning, and coding their games. The games they invented varied between adding digital aspects to traditional outdoor play (e.g. Capture the flag, Hot potato, Relay race) and inventing new games that could not be played without the digital aspects (e.g. forming a 'network' of players with the devices). These effects were evident in both platform versions (coding blocks with/without parameter manipulation). This suggests that the different level of coding flexibility wasn't a barrier for programming outdoor games with the system. Below is a detailed analysis of the findings according to three categories: children's ability to code for outdoor play, enriching outdoor play, and physical manifestation of code in outdoor play context.

Coding for outdoor play

Children invented game ideas relevant for the outdoor play context and were able to use the coding platform to program code that matched their ideas. This activity was both challenging and engaging for the children: they invested a significant amount of time inventing ideas for their games even when they encountered difficulties, with some children spending almost an hour until they reached the desired code for their game (mean=35.29, SD=11.34). Children performed many iterations (mean=20.33, SD=8.82), testing how their code is manifested in the physical world, and exploring various ways to integrate it with outdoor play characteristics.

All children were able to code games for outdoor play, with both levels of Correspondence. 16/24 children achieved Full Correspondence and 8/24 children achieved Partial Correspondence. Partial Correspondence included Game Rules that were not stated in the code (1 child), or code configurations that were not stated in the Game Rules (7 children). Hence, most Partial Correspondences were due to documentation errors, where children left out some of the existing code configurations when they documented the Game Rules. See Figure 5 for an example of a game from each level.

Another indication of children's ability to code for outdoor play was the different ways children integrated the programmable device into their games. Thematic analysis of the Game Rules revealed 3 different roles children assigned to the programmable device:

Game facilitator role (8/24): children programmed the device to digitally facilitate different aspects of the game. Examples include: counting strikes, indicating time, keeping score, and signaling success.

Game objective role (13/24): children programmed the device to take a key role in the game, meaning the goal of the game involved making a deliberate action with the programmable device. Examples include: a Treasure hunt in which the programmable device was used as the treasure and an LED animation signaled it was found; Hot potato in which the device was programmed to set off an animation indicating a strike whenever the timer reached the predefined limit; and a Relay race in which the device was programmed

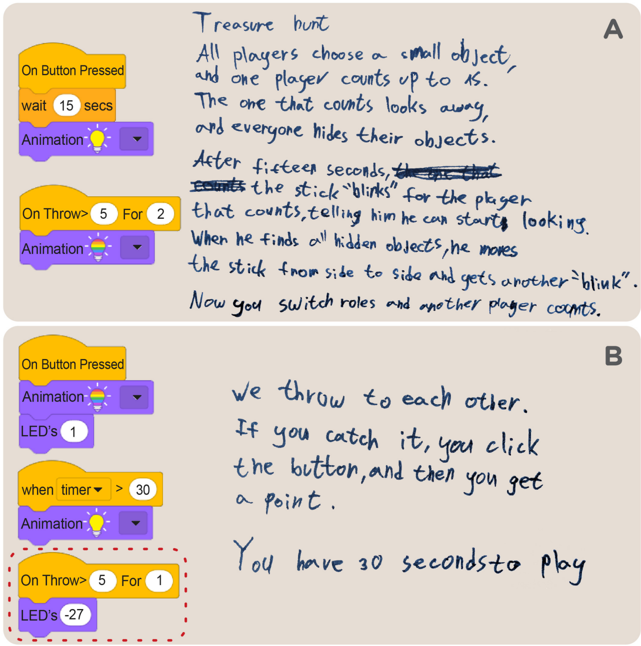

Figure 5: Examples for the different correspondence levels between Game Rules and code configurations invented by children. Example A shows Full Correspondence, and example B shows Partial Correspondence. Both game text descriptions were translated to English by a child, to preserve a child's handwriting.

to be an augmented baton that signals with reduction of score if a pass was illegal.

New game dimension role (3/24): children created a "network" by utilizing several devices at once with matching code to communicate game information between the devices. This type of games is not playable without the digital communication capabilities. For example, one child modified the game Cops and robbers so that each robber had a device. The LEDs were programmed to indicate the number of "live" robbers in the game. When a robber was caught, he/she pressed the button, and all other robbers' devices received the "message" and reduced 1 LED from the "live" robbers count. This enabled robbers to know at all times how many of them are still in the game, even if not all players were in sight.

Enriching outdoor play

Children enriched outdoor play by adding digital aspects that either modified a well-known outdoor game or introduced a novel one. 15/24 children modified well-known outdoor games such as Hide-and-seek, Tag, and Capture the flag. In these cases, the programmable device was used in all three device roles. For example, one child invented a new version of Capture the flag: players divide into two teams and each team hides a device in the play area. After hiding the devices, both teams' players go into the play area one-by-one and look for the other team's device. When a player finds the opposite team's device, instead of grabbing it and running (as in the traditional game), they press the button, earning their team a point (represented by an LED), and the next player can start looking for the device. The team that accumulates all points first, wins (P.11, 10 years old).

7/24 children invented a novel outdoor game that could not be played without the programmable device. In these games, the programmable device was assigned a "Game objective" role. For example, one child invented a game in which players throw their device as far as they can, setting off an LED animation 2 seconds after the device was thrown. The game objective was to run to the device, press the button within 10 seconds of the LED animation, and earn a point (represented by an LED count). If players don't reach the device in time, they lose a point (P.19, 10 years old; see Figure 6).

2/24 children created games that were similar to the games presented to them in the introduction stage of the study (see Figure 3). For example, the goal of one game was to cross the play area without striking out (losing all 5 LEDs). Players begin the game with 5 LEDs on their device, and the LEDs were programmed to decrease if any of the devices exceed a predefined acceleration threshold (P.8, 11 years old).

During the group interviews, children gave various explanations to how their code enriched outdoor play. Some mentioned the additional digital layer: "*This is sort of augmented reality, Hide-and-seek is a game in the real world and here, there is also a digital layer*" (P.5, 9 years old). Others referred to the "objective nature" of technology: "*It* [sensing technology] *makes the game fair. It can solve the problem of over-competitiveness and cheating. If we'd play the same game with an ordinary ball, players could maybe cheat. The device solves these problems as it provides an accurate, objective measure*" (P.16, 11 years old). In addition, children were excited about the possibility of adding digital sensing aspects to outdoor play: "*Its cool because it detects you and opens new possibilities that do not exist with objects we usually play with*" (P.17, 10 years old).

Physical manifestation of code in outdoor play context

The system offered a novel opportunity for physical manifestation of abstract code. 18/24 children associate their code configuration with a physical or social property. Some merged computational concepts with physical game elements, and integrated physical outdoor properties as variables in their code. Others merged computational concepts with social aspects. 6/24 children used the digital device as an external aid to their outdoor games. 15/24 children integrated sensing features as game input and invented games based on the outdoor physical information sampled by the motion sensor. For example, one game's goal was for players to throw and catch the device as many times possible within 10 seconds. Each throw was identified by the accelerometer sensor and gave the player one point (represented by 1 LED). The game ends after 10 seconds, so players can see how many passes they made, and try to beat their score on the next round (P.2, 9 years old). 3/24 children used the devices to communicate information such as how many players have stricken out of the game, even if not all players are visible (as commonly happens in outdoor games). For example, in a modified version of Hide-and-seek, every player has a device and the seeker can count and keep track of the number of players they find by pressing the button on the device and accumulating LEDs. Players that are still active in the game can see how many players are still hiding (P.17, 10 years old). 6/24 children programmed the device as an external aid to the game without direct manifestation of the code in the physical or social outdoor context. In most of these cases, the information was either for keeping score or a timer indicating how much time was left for the game. For example, using the device in a running contest in which each player has a device and can accumulate a point every time they are first to reach the finishing line (P.6, 10.5 years old).

The group interviews revealed that children who associated computational concepts with physical and social game elements were highly enthusiastic about it. For example, one child described the experience as "*seeing your code become a reality*" (P.12, 10.5 years old). Several children specifically mentioned the difference between the standard manifestation of code on the screen and the unique manifestation of code in a physical game, stating that it is in the "*real world instead of on the computer screen*" (P.5, 9 years old), and that they can "*see it and feel it*" (P.24, 10 years old). They also referred to the manifestation of the code in a social context of outdoor play, stating that "*it is outside and it's fun because you program for playing with friends*" (P.10, 9 years old). Some children expressed that programming the physical device facilitated understanding of the logic behind the coding "*Programming for the device made it easier* [than programming for the screen], *everything naturally came together*" (P.4, 10 years old). Some mentioned the physical nature of the debugging process with the system: "*you update the code and you see the outcome on the device, you can easily see what should be changed or fixed*" (P.13, 11 years old).

7 Discussion

In this study, we evaluated if children are able to code for outdoor play, and specifically if they can create their own technologically-enhanced outdoor play experiences. Our findings show that children are able to express their ideas through coding and play their invented games with their friends. Children use code to add a "digital layer" to traditional outdoor games (e.g. Hide-and-seek, Tag, or Capture the flag) and invent novel games. The modular nature of coding allows children to create many different code configurations, used to appropriate the programmable device to their own ideas. Children used the system as an artifact that provides some structure for the play activity, but does not prescribe a fixed mode of engagement. All children used the coding platform as an expressive medium and created personally-meaningful outdoor experiences. In addition, most children leveraged the social and physical aspects of outdoor play in their code (e.g. using the acceleration sensor to detect movement and link it with the players' score), and used them as game elements through the code (e.g. created a communication method between players using a network of devices). Children were excited about the code's manifestation in the physical world and discussed how different it is from the more standard manifestation of code on a computer screen.

Figure 6: Children play a novel game they invented with the system (children photographed with permission).

Children's ability to program the device and use it as an integral part of their outdoor play experiences is not trivial. The essence of outdoor play is "heads up" social interaction and physical activity [11, 16, 31, 43], which are not natural characteristics of coding activities. Moreover, adding screens to outdoor play (a coding platform running on a tablet device) is thought to interfere with natural play patterns [34] and compromise the benefits of natural play [44]. Nonetheless, the children successfully bridged this gap, using the "heads down" coding activity to create code that generates a "heads up" outdoor game experience. In addition, children played their games with their friends with no further adjustments to the code, indicating that they were able to program playable outdoor games. This may be attributed to the clear distinction between the coding platform and the digital device, that framed the coding platform as a tool for defining the game and the device as a tool for playing the game. Future work should test this assumption and compare the digital coding platform to a more tangible coding platform. While physical programming may preserve the "heads up" nature of outdoor play, it may also be perceived as part of the game instead of a mechanism for defining the game rules.

Coding for outdoor play introduces a distinct separation between two known aspects of rules in games: operational rules and constitutive rules [42]. Operational rules are the game instructions defining how to play the game, and constitutive rules define the underlying logical structure of the game. Children have extensive experience with operational rules but are rarely exposed to the constitutive rules of a game. When coding for outdoor play, children had to think about both of them, defining operational rules as the game instructions and creating corresponding constitutive rules using the code. Thus children had to identify the relevant logical structure of the game in order to create an accurate code. All children were able to overcome this non-trivial challenge; they were able to understand the underlying structure of their game and to represent it with code.

The coding platform empowered children to become HUG designers, inventing and implementing their own technology-enhanced outdoor play experiences. Coding extends the possibilities of ready-made (non-programmable) digital outdoor play devices, giving children a way to not only play but also express themselves and invent new physical and social experiences. Children specifically mentioned they felt capable of creating technologically-enhanced outdoor experiences: "It's like making your own game: you make it however you want, and you go outside with the game that you wanted and made, instead of buying a game". In addition, the coding platform wasn't seen as an obstacle or a limitation, but was rather perceived as an opportunity to create: "When I'm playing and suddenly have an idea for a game, I can make it".

Children were also excited about the opportunity to merge the physical and digital worlds in meaningful ways. Most children were able to use physical elements of outdoor play as variables in their code and used it to generate a digital action based on physical events in the real world. The result was a "physical manifestation" of their code in an outdoor play context. Children found this integration of code with the physical world to be playful and exciting, and explicitly mentioned they enjoyed "experiencing their code in the real world".

Limitations

The present study has several limitations. We performed the study with children that are familiar with Scratch, which may limit generalization. In addition, the children were from a mid-high socioeconomic status, some participate in after-school coding classes, and some are very knowledgeable about technology as they follow technology channels on YouTube. In addition, the amount of participating boys and girls in the study (16 boys and 8 girls) wasn't balanced and reflects the gender imbalance in children's after-school coding classes in the authors' country. Clearly, future work should include children with no prior experience with Scratch, and provide them with a sufficient introduction to coding before introducing the system. Our choice to focus the study on children with basic understanding in coding was due to the growing popularity of programming classes in schools. In the near future, many children will have a basic understanding of coding, making our participant sample more relevant to children's general population. Our coding platform has a limited number of coding primitives (blocks), and can not be treated as a full programming language. We decided to focus on 20 blocks to strike the balance between a potentially large number of possible configurations, and a relatively small number of blocks, to not overwhelm children. Another limitation is the two pre-programmed games used in the introduction of the evaluation study, which may have influenced children's ideas when asked to create their own games. However, 22/24 children created a new or modified game that was different from the introduction games. Only two children invented games that seemed to be primed by the introduction.

8 Conclusion

This work indicates that children are able to use code for creating outdoor play experiences, merging abstract code with the physical world in a meaningful and expressive way. Furthermore, children successfully iterated between the "heads down" screen-based nature of coding and the "heads up" nature of outdoor play, without compromising outdoor play characteristics. Children were engaged in creating their own technologically-enhanced outdoor games and were excited to see their ideas come to life through code. We hope our work can motivate industry practitioners and educators to provide children with tools and opportunities to become their own experience creators and use code to enrich their outdoor play experiences based on their own ideas.

9 Acknowledgments

We would like to thank Iddo Wald, Andrey Grishko, Tom Hitron, Nadav Viduchinsky, Yuval Rubin, Rotem Landesman, and Adam Agassi for their great contribution to the project. This research was supported by the Scratch Foundation and by the I-CORE ISF grant #1716/12.

References

[1] Habibe Acar. 2014. Learning environments for children in outdoor spaces. *Procedia-Social and Behavioral Sciences* 141 (2014), 846–853.

[2] Tetske Avontuur, Rian De Jong, Eveline Brink, Yves Florack, Iris Soute, and Panos Markopoulos. 2014. Play it our way: customization of game rules in children's interactive outdoor games. In *Proceedings of the 2014 conference on Interaction design and children*. ACM, 95–104.

[3] Jon Back, Caspar Heeffer, Susan Paget, Andreas Rau, Eva Lotta Sallnäs Pysander, and Annika Waern. 2016. Designing for Children's

Outdoor Play. In *Proceedings of the 2016 ACM Conference on Designing Interactive Systems*. ACM, 28–38.
[4] Jon Back, Elena Márquez Segura, and Annika Waern. 2017. Designing for transformative play. *ACM Transactions on Computer-Human Interaction (TOCHI)* 24, 3 (2017), 18.
[5] Fiona Bailey and Magnus Moar. 2003. The VERTEX project: Designing and populating shared 3D virtual worlds in the primary (elementary) classroom. *Computers & Graphics* 27, 3 (2003), 353–359.
[6] Saskia Bakker, Panos Markopoulos, and Yvonne de Kort. 2008. OPOS: an observation scheme for evaluating head-up play. In *Proceedings of the 5th Nordic conference on Human-computer interaction: building bridges*. ACM, 33–42.
[7] Valerie Barr and Chris Stephenson. 2011. Bringing computational thinking to K-12: what is Involved and what is the role of the computer science education community? *Acm Inroads* 2, 1 (2011), 48–54.
[8] Tilde Bekker and Janienke Sturm. 2009. Stimulating physical and social activity through open-ended play. In *Proceedings of the 8th international conference on interaction design and children*. ACM, 309–312.
[9] Tilde Bekker, Janienke Sturm, and Berry Eggen. 2010. Designing playful interactions for social interaction and physical play. *Personal and Ubiquitous Computing* 14, 5 (2010), 385–396.
[10] Tilde Bekker, Janienke Sturm, Rik Wesselink, Bas Groenendaal, and Berry Eggen. 2008. Interactive play objects and the effects of open-ended play on social interaction and fun. In *Proceedings of the 2008 International Conference on Advances in Computer Entertainment Technology*. ACM, 389–392.
[11] Gabriela Bento and Gisela Dias. 2017. The importance of outdoor play for young children's healthy development. *Porto Biomedical Journal* 2, 5 (2017), 157–160.
[12] Staffan Björk, Jennica Falk, Rebecca Hansson, and Peter Ljungstrand. 2001. Pirates! Using the Physical World as a Game Board.. In *Interact*, Vol. 1. 423–430.
[13] Richard E Boyatzis. 1998. *Transforming qualitative information: Thematic analysis and code development*. sage.
[14] Roger Caillois. 2001. *Man, play, and games*. University of Illinois Press.
[15] Thomas Chatzidimitris, Damianos Gavalas, and Vlasios Kasapakis. 2015. PacMap: transferring PacMan to the physical realm. In *Internet of Things. User-Centric IoT*. Springer, 139–144.
[16] Rhonda Clements. 2004. An investigation of the status of outdoor play. *Contemporary issues in early childhood* 5, 1 (2004), 68–80.
[17] W Clocksin and C Mellish. 2003. Programming in Prolog Using the ISO Standardd, 5< sup> th</sup> ed.
[18] Jill Denner, Linda Werner, and Eloy Ortiz. 2012. Computer games created by middle school girls: Can they be used to measure understanding of computer science concepts? *Computers & Education* 58, 1 (2012), 240–249.
[19] Betsy James DiSalvo, Mark Guzdail, Tom Mcklin, Charles Meadows, Kenneth Perry, Corey Steward, and Amy Bruckman. 2009. Glitch Game Testers: African American Men Breaking Open the Console.. In *DiGRA Conference*.
[20] Andrea A DiSessa. 2001. *Changing minds: Computers, learning, and literacy*. Mit Press.
[21] Kenneth S Goldstein. 1971. Strategy in counting out: An ethnographic folklore field study. *The study of games* (1971), 167–178.
[22] Henry Gorman and Lyle E Bourne. 1983. Learning to think by learning LOGO: Rule learning in third-grade computer programmers. *Bulletin of the Psychonomic Society* 21, 3 (1983), 165–167.
[23] Peter Gray. 2011. The decline of play and the rise of psychopathology in children and adolescents. *American Journal of Play* 3, 4 (2011), 443–463.
[24] Hackaball 2018. Hackaball. Retrieved September 27, 2018 from http://www.hackaball.com/
[25] Koen Hendrix, Guo Yang, Dirk Van De Mortel, Tim Tijs, and Panos Markopoulos. 2008. Designing a head-up game for children. In *Proceedings of the 22nd British HCI Group Annual Conference on People and Computers: Culture, Creativity, Interaction-Volume 1*. British Computer Society, 45–53.
[26] Tom Hitron, Idan David, Netta Ofer, Andrey Grishko, Iddo Yehoshua Wald, Hadas Erel, and Oren Zuckerman. 2018. Digital Outdoor Play: Benefits and Risks from an Interaction Design Perspective. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, 284.
[27] Linda Hughes. 1983. Beyond the rules of the game, why are rooie rules nice. *The game design reader: A rules of play anthology* (1983), 504–516.
[28] James H Humphrey. 2012. *Child development through sports*. Routledge.
[29] TF Juster, F Stafford, and H Ono. 2004. Major changes have taken place in how children and teens spend their time: Child development supplement. *Ann Arbor, MI: Institute for Social Research, University of Michigan* (2004).
[30] Abigail R Kimbell, Andrea Schuhmann, and Hutch Brown. 2009. More kids in the woods: Reconnecting Americans with nature. *Journal of Forestry* 107, 7 (2009), 373–377.
[31] Carsten Magerkurth, Adrian David Cheok, Regan L Mandryk, and Trond Nilsen. 2005. Pervasive games: bringing computer entertainment back to the real world. *Computers in Entertainment (CIE)* 3, 3 (2005), 4–4.
[32] Remco Magielse and Panos Markopoulos. 2009. HeartBeat: an outdoor pervasive game for children. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2181–2184.
[33] Elena Márquez Segura, Annika Waern, Jin Moen, and Carolina Johansson. 2013. The design space of body games: technological, physical, and social design. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*. ACM, 3365–3374.
[34] Pinar Metin. 2003. The effects of traditional playground equipment design in childrenâĂŹs developmental needs. *Yayınlanmamış yüksek lisans tezi. Orta Doğu Teknik Üniversitesi, Ankara* (2003).
[35] Markus Montola, Jaakko Stenros, and Annika Waern. 2009. *Pervasive games: theory and design*. CRC Press.
[36] Netta Ofer, Hadas Erel, Idan David, Tom Hitron, and Oren Zuckerman. 2018. A little bit of coding goes a long way: effects of coding on outdoor play. In *Proceedings of the 17th ACM Conference on Interaction Design and Children*. ACM, 599–604.
[37] Seymour Papert. 1980. *Mindstorms: Children, computers, and powerful ideas*. Basic Books, Inc.
[38] Janet C Read. 2015. Working with Child Participants in Interaction Design. In *Human-Computer Interaction*. Springer, 655–656.
[39] CM Renders, JC Seidell, W Van Mechelen, and RA Hira Sing. 2003. Overgewicht bij kinderen en adolescenten. (2003).
[40] Mitchel Resnick, John Maloney, Andrés Monroy-Hernández, Natalie Rusk, Evelyn Eastmond, Karen Brennan, Amon Millner, Eric Rosenbaum, Jay Silver, Brian Silverman, et al. 2009. Scratch: programming for all. *Commun. ACM* 52, 11 (2009), 60–67.
[41] Mitchel Resnick, Fred Martin, Randy Sargent, and Brian Silverman. 1996. Programmable bricks: Toys to think with. *IBM Systems journal* 35, 3.4 (1996), 443–452.
[42] Katie Salen, Katie Salen Tekinbaş, and Eric Zimmerman. 2004. *Rules of play: Game design fundamentals*. MIT press.
[43] Iris Soute and Panos Markopoulos. 2007. Head up games: the games of the future will look more like the games of the past. In *IFIP Conference on Human-Computer Interaction*. Springer, 404–407.
[44] Iris Soute, Panos Markopoulos, and Remco Magielse. 2010. Head Up Games: combining the best of both worlds by merging traditional and digital play. *Personal and Ubiquitous Computing* 14, 5 (2010), 435–444.

[45] Storyball 2018. Storyball - Keeping Children Active & Playful. Retrieved September 27, 2018 from http://www.mystoryball.com/home

[46] Katie Salen TekinbasÌğ. 2006. *The game design reader: A rules of play anthology*. MIT press.

[47] Jeannette M Wing. 2006. Computational thinking. *Commun. ACM* 49, 3 (2006), 33–35.

[48] Jeannette M Wing. 2008. Computational thinking and thinking about computing. *Philosophical transactions of the royal society of London A: mathematical, physical and engineering sciences* 366, 1881 (2008), 3717–3725.