Using A Visual Programming Environment and Custom Robots to Learn C Programming and K-12 STEM Concepts

Sai Prasanth Krishnamoorthy
Mechatronics, Controls, and Robotics Laboratory
Tandon School of Engineering
New York University
saiprasanth@nyu.edu

Vikram Kapila
Mechatronics, Controls, and Robotics Laboratory
Tandon School of Engineering
New York University
vkapila@nyu.edu

ABSTRACT

This paper presents a robot-Visual Programming Environment (VPE) interface that can support K-12 students to learn science, technology, engineering, and math (STEM) concepts. Specifically, we employ Google's Blockly VPE to construct a blocks-based visual programming tool to facilitate easy programming of and interaction with physical robots. Through a careful and intentional integration of the Blockly VPE and physical robots, we illustrate that many K-12 level STEM concepts, which are traditionally treated through lectures and problem-solving, can be explored in a hands-on manner. The use of Blockly VPE obviates the need for prior experience with computer programming or familiarity with advanced programming concepts. Moreover, it permits students to learn various programming constructs, sequentially, starting from the fundamentals and gradually progressing to advanced concepts. The web-based Blockly VPE provides an interface that allows the user to browse through a block library and construct a block code for which a corresponding C program is automatically generated. The default web-based Blockly interface has been modified to permit the user to edit the resulting C program or to create an entirely new C program. Moreover, the Blockly VPE allows the user to wirelessly upload the C program to a Linux server running on a Raspberry Pi computer hosted on the robot. The Raspberry Pi compiles the received C program and serially transfers corresponding instructions to the robot's embedded hardware. The efficacy of the proposed robot-VPE interface is examined through students' experiences in conducting several illustrative robot-based STEM learning activities. The results of content quizzes and surveys show gains in students' understanding of STEM concepts after participation in robotics activities with the VPE interface.

CCS Concepts

•**Applied computing** → **Interactive learning environments;** *Arts and humanities;* •**Computer systems organization** → **External interfaces for robotics;** •**Human-centered computing** → *Activity centered design;*

Keywords

Blockly; Interface; K-12 STEM Education; Puppet Robot; Robotics; Visual Programming

FabLearn'16, October 14-16, 2016, Stanford, CA, USA

DOI: http://dx.doi.org/10.1145/3003397.3003403

1. INTRODUCTION

Recent years have witnessed tremendous changes in the way K-12 education is conducted [13]. Advancements in STEM continue to drive the need to integrate challenging STEM concepts in the K-12 STEM curriculum [8], [18]. Increasingly, advance concepts, such as computer programming and robotics, are being introduced to students at ever younger ages. Even as computing and robotics technologies engage and enhance student learning, it is of paramount importance to create novel learning tools and environments, which are developmentally appropriate, to render the learning process engaging, entertaining, and streamlined [32]. Thus, introduction of advanced STEM and computing concepts in the K-12 curriculum necessitates development and application of attractive and interest-invoking learning technologies that can engage the students. For example, frequently, toys and games have been used to assist educators in teaching various concepts from diverse disciplines [15], [28]. In a similar vein, robots are increasingly being used in K-12 STEM education by providing hands-on learning experiences that promote student engagement and participation [6], [21].

This paper presents a method to support student learning of K-12 STEM concepts using robotics and a VPE interface. As a byproduct of engaging in learning with robotics, students can build foundational knowledge in computer programming [3]. The concept of using a robotics-VPE interface for K-12 STEM education is well established as evidenced by commercially available LEGO EV3 robotics platform [10]. Nonetheless, the costs of such robotics kits may be prohibitive for some schools and they often necessitate use of a personal computer (PC) with at least medium-level performance capabilities. In contrast, this paper provides an approach to learn a variety of STEM and computing concepts using an open-source VPE interface with low-cost, open-source, robot hardware. Specifically, the two robots used in this paper can be constructed relatively easily at school or home using 3D-printed components. Moreover, the operating system of the robot requires only a lightweight Java-based software environment that can be run on any computer, with a browser installed, including tablets and

Figure 1: System Architecture

mobile devices. Users can also adapt our robot-VPE interface to interact with their pre-existing robots. Affordability and flexibility in adoption makes this system a viable alternative to commercially available educational robotics kits.

2. HARDWARE AND SOFTWARE INTERFACE

The robot-VPE interface features a simple network of electronics and software that can be easily set up and configured. The system is designed to be compatible with custom robots that can be made at school or home with little knowledge about robotics. The primary objective of this work is to prototype, evaluate, and validate an educational robotics tool that can assist students in learning K-12 STEM concepts and basics of computer programming while retaining simplicity in the design and cost-effectiveness of the tool. The two robots used in our study are simple mechatronic systems made using 3D printers and off-the-shelf servo motors and electronics. The software is configured to be compatible with robots that may already exist in schools, making it easy for students and teachers to adopt the VPE interface with these robots. The following subsections describe the hardware and software subsystems along with details of the subsystem integration and control.

2.1 Hardware Environment

The system features two robots, a robotic puppet and a robotic arm, for experimenting with the VPE interface in classrooms. The robotic puppet has been created to engage students from both genders in STEM learning through art. Moreover, the robotic arm has been created to present real-world applications of K-12 science and math concepts in developing solutions to engineering problems. On both robotic platforms, a Raspberry Pi [30] is used to compile the C program sent from the VPE interface and an Arduino embedded microcontroller [4] controls the actuators. Figure 1 shows the system architecture of the robots.

2.1.1 Robotic Puppet

For centuries, civilizations have used puppets as means to educate and positively influence people [5], [7]. Puppets have been used in children's education for varied topics, e.g., to teach healthy practices [2], to raise awareness about certain health conditions [25], to strengthen social and emotional competence in young children [31], to impart cultural traditions and societal values [16], to promote science engagement and discourse [27], and to address misconceptions in mathematics [14]. The success of children's puppet-based educational entertainment shows, e.g., 123 Sesame Street and Bear in the Big Blue House, has paved the way for TV programs and interactive educational toys. In recent years, engineering principles and control theory have been applied to the design of expressive behaviors and motions for puppet robots [20], [34]. Furthermore, toy-based robotic platforms, such as LEGO Mindstorms, have been widely adopted for providing practical learning experiences to students [33]. The elegance of art and expression in puppets can be combined with the popularity and practicality of robotics to produce educational platforms that effectively address learning objectives [36]. Prior research has established that mindful integration of robotics in K-12 education can engage pupils of both genders [12], [22]. Thus, the robotic puppet shown in Figure 2, a combination of robotics and art, can be effectively used to engage diverse audiences to learn about STEM concepts, including those who may be more interested in arts and crafts, dance, or music. The robot's actuation mechanism is a modular system that can be adapted and integrated with any marionette puppet. Specifically, this actuation mechanism consists of 3D-printed robotic arms that impart the robot eight degrees of freedom (DOF) (see Figure 3). The shoulder, forearm, elbow, and knee joints are independently controlled. Each joint is actuated by a DC servo motor with 180°range. The motors are controlled by an Arduino Mega containing an ATMega2560 chip. Moreover, an Arduino Uno microcontroller equipped with a VS1053 audio chip is used for playing audio tracks.

Figure 2: Robotic Puppet

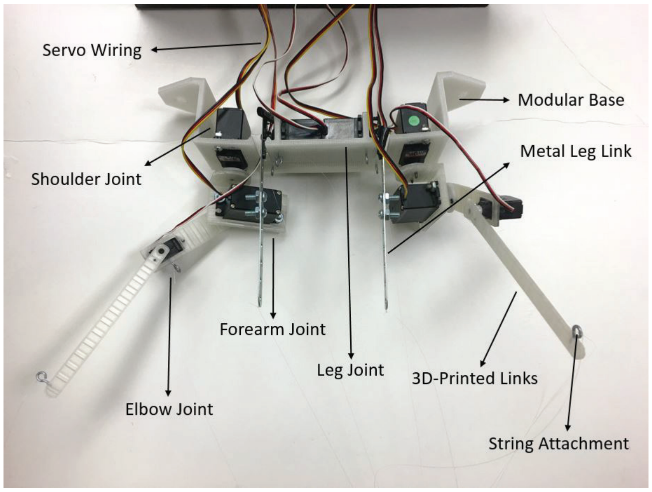

Figure 3: Robotic Puppet Actuation Mechanism

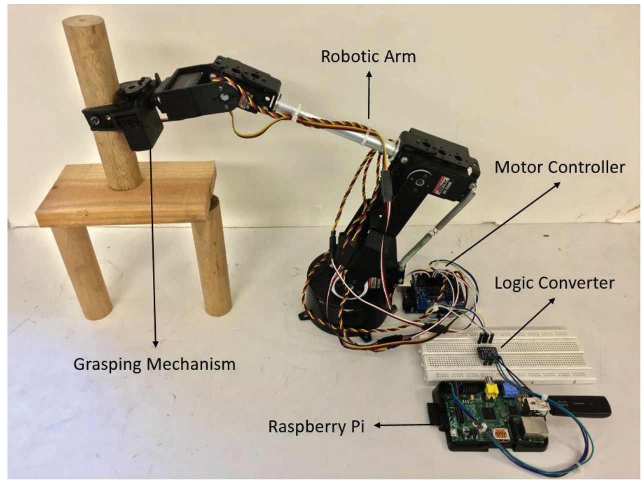

Figure 4: Robotic Arm Performing Bridge Construction Activity

The ATMega2560 board is used as a servo controller and is configured to receive serial commands from the Raspberry Pi. The web-based VPE interface generates the C program for the user created block code and transfers it to the Raspberry Pi via Wi-Fi. The Raspberry Pi compiles and executes the received C program, resulting in instructions being sent to the robotic puppet's servo controller and the audio player board. The robot, including the microcontroller boards and servos, is powered by a 6V, 4A DC supply and the Raspberry Pi is powered by a 5V, 2A DC supply.

2.1.2 Robotic Arm

The robotic arm shown in Figure 4 is a 5 DOF manipulator with a pick and place mechanism. The device is made entirely from off-the-shelf components. The shoulder joint consists of 2 DOF, the elbow joint has 1 DOF, and the wrist joint contains 2 DOF and an additional motor for the grasping mechanism. The arm consists of 6 hobby DC servo motors located at the shoulder, elbow, and wrist joints, with each DOF being actuated by one motor. The 6 servo motors are controlled by an Arduino Uno microcontroller containing an Atmega16u chip. Each joint movement instruction is available as a block in the Blockly-based VPE interface. The C program generated by the user with the robot's movement blocks is transferred to the Raspberry Pi via Wi-Fi. The instructions from the compiled and executed C program are sent to the servo controller for performing the tasks as directed by the user. The entire robotic arm system is powered by a 6V, 4A DC supply.

2.2 Software

Computer programming is increasingly gaining importance in diverse academic fields and careers and with its growing importance students are being introduced to basic programming concepts at early grades [11]. It is expected that such an early exposure to computing will allow students a facility with computational thinking that will promote faster and efficient learning at higher educational levels [17]. There is a growing trend to incorporate C as a common programming language at the high school level. However, care should be taken to prepare and expose students to fundamental programming constructs from early grades so that they can perform advance programming in a text-based environment at the high school level.

Our VPE interface utilizes Google's Blockly programming tool to create a user interface for blocks-based code construction [1], [29]. As C language becomes prevalent in K-12 computing curriculum, we have modified [35] the Blockly interface to generate C program, instead of the Java program that is typically created by Blockly. Our VPE block library contains drag-and-drop blocks that are created to control the robotic puppet and the robotic arm. The Blockly environment generates a C program based on a block code assembled by the user. The current user interface offers over 60 blocks, including distinct movement commands for the robots and various C programming constructs, such as loops, decisions, logical conditions, and functions, among others. From the block code, the VPE interface generates the C program in a parallel tab. Within this tab, the user can modify the generated C program or create a new C program. The C program is then sent via Wi-Fi to the Raspberry Pi located on the robots for compilation. The VPE interface alerts the user if any compilation errors occur. The Raspberry Pi contains libraries to interpret the C program transferred from the VPE interface. Following successful compilation, the program is executed and the Raspberry Pi interacts with the robot microcontrollers serially to relay the instructions from the C program. The users can create custom blocks in addition to those available in the block library. This feature can be potentially used to integrate user-made robots with the VPE. The software application is web-based and can be hosted on any web browser either on a computer or a mobile device. The VPE interface is computationally less intensive and thus can be run on low-end computers. This can significantly reduce the cost of implementation. Figure 5 depicts the screen-shot of the developed VPE interface.

2.3 Cost of Construction

This section discusses the overall costs involved in creating the robots. The software development is based on free

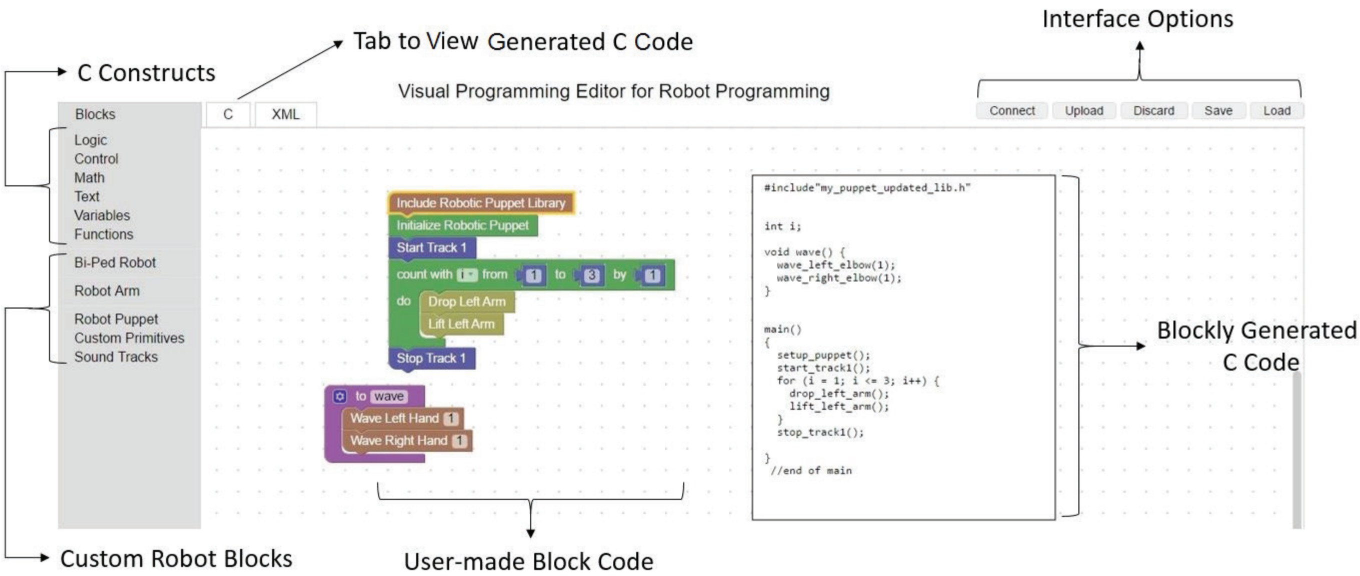

Figure 5: Software Interface

open-source design. The robot puppet is developed using 3D printers. Moreover, the robot arm is constructed using off-the-shelf components. Arduino microcontrollers and Raspberry Pi are being used as controllers for both the robotic devices. The robot joints are actuated by low-cost standard servo motors. Moreover, the robot controllers and servo motors can be reused for multiple robotic projects. Table 1 provides a breakdown of average costs involved in building these robots.

Table 1: Average Cost of Robot

Item	Cost
Robot controllers	$60
Motors	$60
Building material	$60
Miscellaneous	$20
Total	$200

Remark 1: Recent years have witnessed many public libraries and schools establishing their own maker spaces [26], where 3D printers are prevalent [9], [19], and where students can build their project prototypes and perform science experiments. Furthermore, many of these maker spaces have spurred the creation of thriving robotic communities by enthusiastic teachers and hobbyists. Such maker spaces can enable interested students and teachers to collaborate and construct this paper's robotic platforms and utilize the generic software package used in the robot-VPE system.

Remark 2: Although *software-only* educational platforms offer economic and quick implementation, incorporation of a hardware platform helps students to creatively build and use physical devices while making mistakes and learning from them. Moreover, the robotic puppet presents a viable alternative to students who like to explore STEM+Art fields and offers an option to parents who prefer a creative and productive extra-curricular activity for their children. The cost of the proposed educational platform depends on the hardware needed to construct a robot with desired functionality and construction quality. The Blockly-based VPE software package is free to use. While the Blockly-based VPE is similar to other VPEs such as Scratch and App Inventor, Blockly provides a means to generate syntax for various structured programming languages, allowing the students to inspect the underlying code and learn text-based programming as well.

3. K-12 STEM LESSON DESIGN

With the knowledge of K-12 STEM curriculum and standards, specific activities have been adopted to test the robot-VPE setting among three student groups, K-5, 6-8, and 9-12 grades. The developed activities employ both robotic puppet and robotic arm platforms as students learn STEM-related concepts (e.g., distance, measurement, gravity, center of mass, etc.), computer programming, and problem solving skills [23], [24]. These lessons were selected and designed based on the feedback and curricula topics suggested by teachers in each of the target grade levels. The objective of the selected lessons was to evaluate the practical and interactive learning experience gained by students using robots. Future educators can freely choose from a wide variety of STEM concepts and accordingly build robots specifically designed to explore the robot-VPE based practical learning approach.

3.1 Grades K-5 Lessons

The STEM activities designed for elementary school students focus on introduction to robots and its physical capabilities compared to humans. The students are encouraged to inspect the physical characteristics of humans and compare them with the robots. The robotic arm is used as an instrument to explain various physical components of a robot and its similarities and differences compared to human beings. The activity encourages students to inspect toys and other mechanical objects around them and understand how their mechanical components work. In the "Reach" activity, the students are able to inspect their arm's reach while sitting on a chair. They replicate the same procedure to determine the reach of the robotic arm. Students are asked to pick up an object in front of them while analyzing the movements and the trajectory they use to perform the task. Next, they are tasked to make successive movements for individual joints of the robotic arm to perform the same task by manually moving the robot's joints. The students perform the "Distance and Measurement" activity by measuring the height of the robot arm by moving different joints. For the robotic puppet activity, the students inspect their own knee, elbow, and shoulder joints and find the physical reach limitations of each joint. Next, they inspect the joints on the robotic puppet and explore the manner in which these joints are actuated by the robot mechanism compared to their own muscles. Students are tasked to move individual joints of the robotic puppet using block commands in specified sequences, giving them an understanding of the function of the robotic puppet actuators. Next, the "Counting" activity is performed to explore basic addition and subtraction by counting the number of successive movements made by the robotic puppet in the left and right arms. The final activity requires students to perform a paper-design of a robot that can shake hands, drawing inspiration from the previous activities.

3.2 Grades 6-8 Lessons

The middle school students start by exploring the robot mechanism including physical dimensions and range of motion. They begin by moving the joints of the robot arm to specific angles using the blocks-based code and record the difference in the projection of end effector on table (indicated by laser a beam). Next, they are tasked with creating a pick and place movement program using the blocks-based code, where the robot arm is to pick an object at a specific location in its workspace and drop it at another location. Moreover, they are allowed to explore the possibility of picking among multiple objects to create a structure made of plastic blocks. Finally, the "Demolition" activity explores the concepts of measurement, gravity, and center of mass. Here, the students are tasked to topple a small structure while preventing debris from falling out of a safe region. As with the robot arm, the students inspect the robotic puppet anatomy, range of motion, and constraints. They envision scenes from plays and construct a script for the robot puppet while giving playback voice. The robotic puppet activities for the middle school primarily focus on the usage of various programming constructs for creating fluid movements of the puppet.

3.3 Grades 9-12 Lessons

The high school students inspect the construction, mechanism, and physical characteristics of the robot, including DOF, range of motion, and load bearing capabilities. By using a ruler and protractor, they measure and determine robot trajectories that drive the robot arm to specific locations in the workspace and build block code corresponding to the desired motion. For the "Center of Mass" activity, students are instructed to pick and place a uniform plastic beam with uneven weights on either ends, while maintaining a steady motion, by calculating the optimal location for the robotic arm gripper to hold the beam. Next, the "Bridge Construction" activity (see Figure 4) requires the students to use the robotic arm to pick and place block structures and beams to construct a bridge. Students are encouraged to use various programming primitives such as functions, loops, variables, etc., to simulate an automated sequence using a minimal block code while constructing the bridge. As with the robot arm, after inspecting the robotic puppet anatomy, range of motion, and constraints, the students engage in a design challenge to create their own unique designs to actuate the puppet toy. Additionally, the students are tasked to choreograph a dance sequence for the robotic puppet. The activity requires precise timing to synchronize puppet movements with the song. Students are required to use functions, loops, and delays in their block program for streamlined synchronization.

4. ASSESSMENT

To evaluate the extent to which students learned STEM concepts with the robot-VPE system, as well as their level of engagement in the performed activities, this section describes the assessment conducted with a class of 25 students from grades 5–9 who met as a group in a classroom setting. Prior to being introduced to the robotics systems, the students participated in a background test that established their knowledge of prerequisite concepts and their prior learning experiences on the topics of the selected lessons. Following the test, the students were given a comprehensive explanation, along with examples, for the STEM concepts involved in the lesson plans. These lecture sessions paralleled the classroom learning experience of the students. During these instructional activities, the instructor refrained from mentioning the use of robot-VPE system or how the VPE activities utilize the STEM concepts being discussed. Next, the students were administered a pre-test to measure their level of understanding of the STEM concepts from the lecture session. The test consisted of questions covering the discussed STEM concepts along with simple numerical problems to test the students' problem-solving skills and applied understanding. Following the test, the students were given an hour long demonstration on using the VPE interface. The students were encouraged to practice the block program construction using tutorials specifically designed to simplify the introduction. Following the demonstration, the students were introduced to the idea of being able to control the robots using the VPE interface. Each student independently experimented with the robot-VPE system to gain hands on learning experience in the previously discussed STEM concepts. The students also associated the generated C program with the robot's movements and examined the flow of the C program. The students were able to associate the C programming constructs with the robot movements to understand the motivation for using each C statement. Following the hands-on, visual programming and experimentation activities, the students were administered a post-test to check their understanding of the relevant STEM concepts. The post-test had questions similar to the pre-test, including some additional applications-based questions. The tests taken by the students during the activities included multiple-choice, numeric, and true/false questions. Each student's test score was scaled to 10 and constituted the final grade. Figure 6 shows the sequencing of these assessment activities. The robotics-VPE system was additionally demonstrated to over 150 participants from various grade levels, parents, teachers, and researchers. Individually, each participant tested the robot-VPE interface and choreographed a simple story-line for the robot puppet using various programming constructs with the block code. Among this group, 45 K-12 students answered a survey.

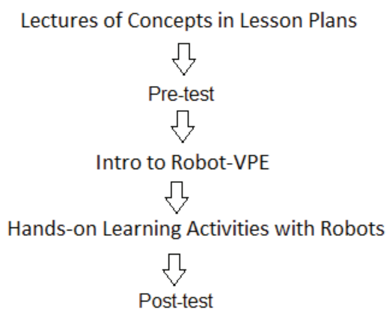

Figure 6: Assessment Sequencing

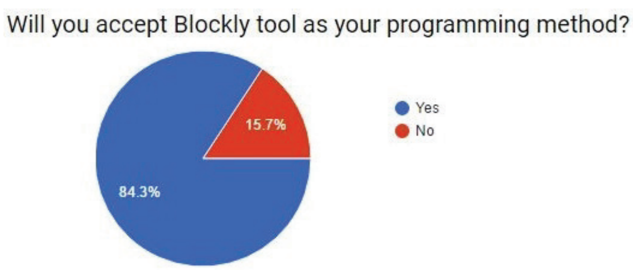

Figure 7: Students' Opinion

4.1 Observations and Results

The response to the robot-VPE system was largely positive and the students, teachers, and parents who tested the interface welcomed the idea of introducing lesson plans featuring visual programming using robots. Figure 7 shows the opinion of 70 K-12 students regarding the adoption of Blockly VPE in classrooms to learn programming. Although the learning experience using robot-VPE interface is not a direct replacement to the traditional learning practices, it is a viable alternative to hands-on learning activities that augment the understanding of the concepts the students learn in the class. The following observations are based on 25 classroom and 45 demonstration group K-12 students, with evenly split participation of both genders, as shown in Figure 8. Figure 9 depicts the pre-/post-test results for the 25 students from the classroom group. The one standard deviation is indicated by the vertical lines on each bar graph.

The pre-/post-test results show increase in students' performance following participation in the robot-VPE learning activities. The robotic arm activities showed a 26% increase in correct answers, while the robotic puppet activities showed over a 60% increase. Girls participating in the post-test after robotic-puppet activities correctly answered twice the number of questions compared to the pre-test, demonstrating that interest invoking tools engage students from both genders while learning STEM concepts. Next, Figure 10 shows the cumulative responses to the survey from the 70 K-12 students. Over 70% of the students understood the flow of the C program by associating it with the movement pattern of the robots. Also, over 90% of the students responded that the robot-VPE interface is engaging. The students were excited by the prospect of gaining hands-on experience using robots and programming, even as almost 80% of the students had prior experience using robots for play or education. The feedback from the students on the current learning methods employed by the schools were mixed, with 50% of the students maintaining either unfavorable or neutral opinion. This reinforces the idea to utilize robot-VPE interface to augment and assist classroom learning environment while providing engaging and practical hands-on learning experience to students. Based on the feedback from the teachers, a majority of the schools are equipped with computers for running Blockly-based VPE and have access to robotic kits for hands-on learning. There is an immediate requirement for systems like the robot-VPE interface, which can convert the existing robotic platforms into tools for learning using hands-on activities while requiring minimal additional resources. Installing robot-VPE on existing platforms will be relatively cheap and easy to customize compared to providing commercial education kits to each student.

5. CONCLUSIONS AND FUTURE WORK

The robot-VPE interface is a platform for students to interactively experiment and learn K-12 STEM concepts by using robotic systems, such as the robotic puppet and the robotic arm. The VPE interface is constructed using Google's Blockly framework and its light-weight architecture allows the VPE to run on any browser on computers or mobile devices, thus needing minimal system requirements. The robotic platforms are constructed based on open-source designs and are actuated using hobby DC servo motors and controlled using Arduino and Raspberry Pi boards. The equipment used in the construction of the robots is economic and easily available, making the proposed robot platforms efficient and affordable compared to commercially available robotic kits. The designed VPE interface can also be configured to run with existing robotic platforms. The lesson plans presented to the students and the pre-/post-assessments suggest that the students gained increased understanding of the concepts covered by using hands-on learning approach with robot-VPE interface. Moreover, from the participant feedback, it is observed that the system has been successful in engaging a wide variety of audiences regardless of gender and age, making it suitable for deployment in classrooms. Future work will consider comparing the effectiveness of the robot-VPE system with software-based platforms and other low-cost hardware platforms. Furthermore, additional research will be conducted to investigate mechanisms for seamless integration of the proposed platform in formal classroom setting. Based on the feedback provided by teachers and home-schooling parents, we intend to create a remote access system to allow students to remotely log-in into the robot using any web browser. The students will be able to use the web interface to program the robot using the VPE and upload the C program. A camera setup will allow them to view the robot performing instructed movement patterns. This system will enable students lacking access to custom robots to gain hands-on learning experience remotely.

6. ACKNOWLEDGMENTS

This work is supported in part by the National Science Foundation awards RET Site EEC-1542286, EEC-1132482, and DRK-12 DRL: 1417769, and NY Space Grant Consortium grant 76156-10488.

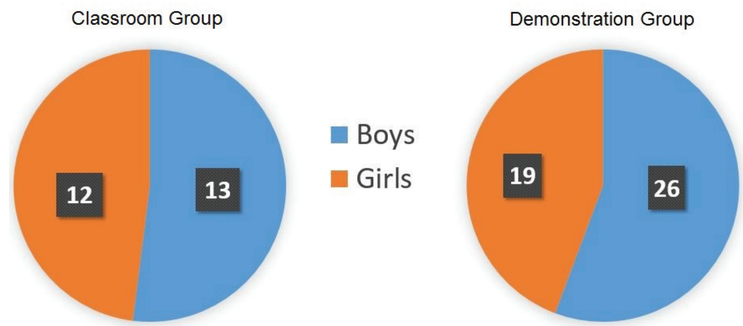

Figure 8: Participant Gender Demographics

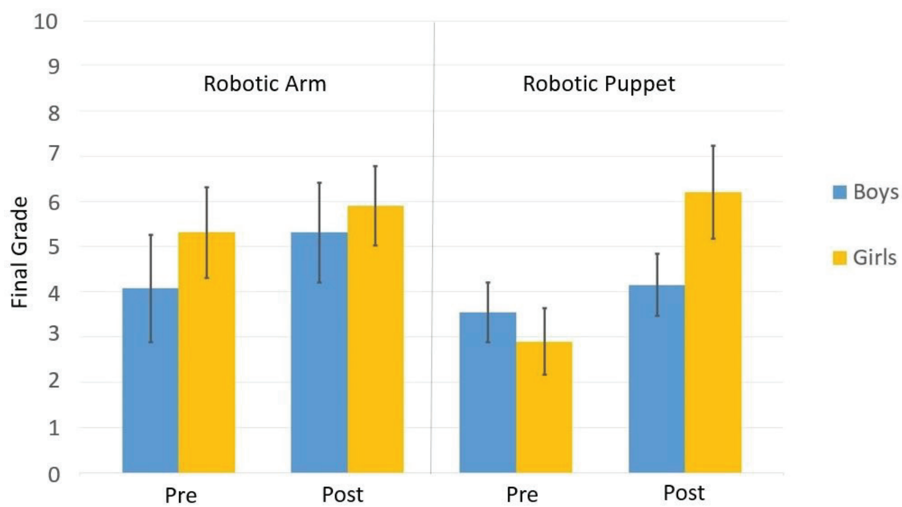

Figure 9: Performance Assessment

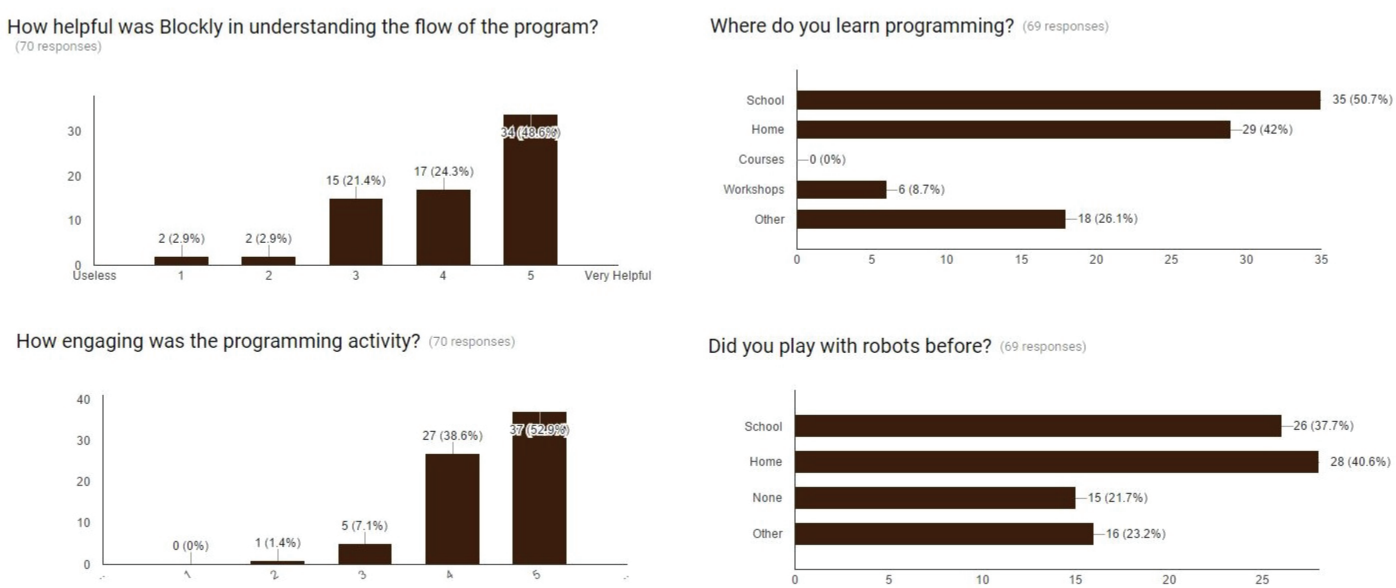

Figure 10: Survey Results

7. REFERENCES

[1] Google for education: Blockly. *URL: https://developers.google.com/blockly/*, 2016.

[2] M. Bakhit *et al*. Puppets in education. Public Health Projects, 2011.

[3] T. Balch *et al*. Designing personal robots for education: Hardware, software, and curriculum. *IEEE Pervasive Computing*, 7(2):5–9, 2008.

[4] S. Barrett. *Arduino Microcontroller Processing for Everyone!* Morgan & Claypool, 2013.

[5] J. Bell. *Strings, Hands, Shadows: A Modern Puppet History*. Wayne State University Press, 2000.

[6] F. Benitti. Exploring the educational potential of robotics in schools: A systematic review. *Computers & Education*, 58(3):978–988, 2012.

[7] M. Bernier and J. O'Hare (editors). *Puppetry in Education and Therapy: Unlocking Doors to the Mind and Heart*. Bloomington, IN: AuthorHouse, 2005.

[8] S. Brophy *et al*. Advancing engineering education in P-12 classrooms. *Journal of Engineering Education*, 97(3):369, 2008.

[9] E. Canessa, C. Fonda, and M. Z. (editors). *Low–cost 3D printing for science, education and sustainable development*. ICTP, 2013.

[10] B. Erwin, M. Cyr, and C. Rogers. LEGO engineer and ROBOLAB: Teaching engineering with LabVIEW from kindergarten to graduate school. *Int. Journal of Engineering Education*, 16(3):181–192, 2000.

[11] G. Fessakis, E. Gouli, and E. Mavroudi. Problem solving by 5–6 years old kindergarten children in a computer programming environment: A case study. *Computers & Education*, 63:87–97, 2013.

[12] S. Hartmann, H. Wiesner, and A. Wiesner-Steiner. Robotics and gender: The use of robotics for the empowerment of girls in the classroom. In *Gender Designs IT*, pages 175–188. Springer, 2007.

[13] M. Herk. Big changes for K12 education. Committee for Economic Development, 2015.

[14] B. Keogh and S. Naylor. Puppets count. *Mathematics Teaching*, 213:32–34, 2009.

[15] J. Kirriemuir and A. McFarlane. Use of computer and video games in the classroom. In *Conf. of Digital Games Research Association (DIGRA)*, 2003.

[16] M. Kruger. Puppets in education and development in Africa: The puppet's dual nature and sign systems in action. *South African Theatre Journal*, 21(1):64–74, 2007.

[17] S. Kurebayashi, T. Kamada, and S. Kanemune. Learning computer programming with autonomous robots. In *In. Conf. on Informatics in Secondary Schools-Evolution and Perspectives*, pages 138–149. Springer, 2006.

[18] A. Linson and R. Lamon. Enhancing K-12 curriculum with STEM integration. *Improving the Quality of STEM Education*, 2015.

[19] H. Lipson and M. Kurman. *Fabricated: The new world of 3D printing*. John Wiley & Sons, 2013.

[20] P. Martin, E. Johnson, T. Murphey, and M. Egerstedt. Constructing and implementing motion programs for robotic marionettes. *IEEE Transactions on Automatic Control*, 56(4):902–907, 2011.

[21] M. Mataric. Robotics education for all ages. In *Proc. AAAI Spring Symp. on Accessible, Hands-on AI and Robotics Education*, 2004.

[22] E. Milto, C. Rogers, and M. Portsmore. Gender differences in confidence levels, group interactions, and feelings about competition in an introductory robotics course. In *Proc. of Frontiers in Education*, volume 2, pages F4C–7, 2002.

[23] S. Norton, C. McRobbie, and I. Ginns. Problem solving in a middle school robotics design classroom. *Research in Science Education*, 37(3):261–277, 2007.

[24] A. Ortiz. *Fifth grade students' understanding of ratio and proportion in an engineering robotics program*. Ph.D. Dissertatio, Tufts University, 2010.

[25] J. Pélicand *et al*. A therapeutic education programme for diabetic children: Recreational, creative methods, and use of puppets. *Patient Education and Counseling*, 60(2):152–163, 2006.

[26] K. Sheridan *et al*. Learning in the making: A comparative case study of three makerspaces. *Harvard Educational Review*, 84(4):505–531, 2014.

[27] S. Simon *et al*. Puppets promoting engagement and talk in science. *Int. J. of Science Education*, 30(9):1229–1248, 2008.

[28] K. Squire and H. Jenkins. Harnessing the power of games in education. *Insight*, 3(1):5–33, 2003.

[29] J. Trower and J. Gray. Blockly language creation and applications: Visual programming for media computation and bluetooth robotics control. In *Proc. 46th ACM Tech. Symp. on Computer Science Education*, page 5, 2015.

[30] E. Upton and G. Halfacree. *Raspberry Pi User Guide*. John Wiley & Sons, 2014.

[31] C. Webster-Stratton and M. Reid. Strengthening social and emotional competence in young children - The foundation for early school readiness and success: Incredible years classroom social skills and problem-solving curriculum. *Infants & Young Children*, 17(2):96–113, 2004.

[32] L. Whitman and T. Witherspoon. Using LEGOs to interest high school students and improve K12 STEM education. *Change*, 87:76, 2003.

[33] K. Williams *et al*. Enriching K-12 science and mathematics education using LEGOs. *Advances in Engineering Education*, 3(2), 2012.

[34] S. Xing and I.-M. Chen. Design expressive behaviors for robotic puppet. In *Int. Conf. Control, Automation, Robotics and Vision*, volume 1, pages 378–383, 2002.

[35] R. Yadagiri, S. Krishnamoorthy, and V. Kapila. A blocks-based visual environment to teach robot-programming to K-12 students. In *Proc. Amer. Soc. Eng. Ed.*, Seattle, WA, 10.18260/p.23358, 2015.

[36] H. Yanco, H. Kim, F. Martin, and L. Silka. Artbotics: Combining art and robotics to broaden participation in computing. In *Proc. AAAI Spring Symposium on Robots and Robot Venues: Resources for AI Education*, page 192, 2007.