



A summer expedition to robotics

Weihua Sheng¹, Yujiang Xiang¹, Zhidong Su¹, Fei Liang¹,
Eric Wang, Albert Sheng, and Charles Liu



This article documents the science, technology, engineering, and mathematics (STEM) robotics summer camp held in June 2022,

which is part of the outreach activities of two National Science Foundation-funded projects at Oklahoma State University. This STEM summer camp aims to address the challenges in organizing robot-centered STEM summer camps for students in kindergar-

ten through 12th grade (K-12). We explored various new ways to prepare and conduct the summer camp.

First, an age-appropriate competitive game was developed to engage middle school and high school students in the camp activities. This game features a jungle expedition

Digital Object Identifier 10.1109/MPOT.2023.3247909
Date of current version: 5 May 2023

competition that involves the construction of a remotely operated mobile robot and Python programming in a team environment. Second, a novel trivia game system was developed, which is also part of the hands-on project for the students, to motivate student learning and assess the learning outcomes in a highly entertaining and interactive fashion.

In addition, this summer camp was prepared and organized with contributions from a wide variety of students, including graduate students, undergraduate students, and high school students, which ensures that the program best fits the age groups without losing its rigor. It is our belief that STEM summer camps that are fun, engaging, and rigorous will have long-term impacts on K–12 students as technologies such as robotics, artificial intelligence (AI), autonomous driving, and Internet of Things (IoT) are rapidly changing our lives.

Introduction

Background

With the rapid evolution of technologies, including robotics, AI, autonomous vehicles, and the IoT, the United States is facing a shortage in the STEM workforce (National Science Foundation, 2023). Such a shortage can be addressed through greater participation by underrepresented groups, particularly minority groups and women, in the STEM workforce (Diekmann and Benson-Greenwald, 2018). Therefore, there is a great need to attract K–12 students into STEM fields, especially those from underrepresented groups. This workforce shortage problem is also prevailing in many other major developed countries and developing countries (Mulvey et al., 2022).

Personal interest and motivation are key components in inspiring students to pursue careers and paths in STEM learning. Students' interest and motivation contribute to their success in learning and retaining STEM content. Summer camps are effective ways of providing K–12 students the

Educational robots have been used in many summer camps in recent years and have proven to be valuable in motivating students to learn STEM topics.

opportunities to learn knowledge and skills in STEM (Mohr-Schroeder et al., 2014). Among the many important skills that many future STEM jobs require are the problem-solving skill and the programming skill. Educational robots have been used in many summer camps in recent years and have proven to be valuable in motivating students to learn STEM topics (Balaguer-Álvarez, 2017).

Objectives

This article aims to address two major challenges in these robot-centered STEM summer camps: 1) how to engage students and maximize their interest in the summer camp activities and 2) how to effectively motivate student learning and assess learning outcomes during the summer camps.

Regarding the first challenge, research has found that cooperative games help improve learning effectiveness in students (Creighton and Szymkowiak, 2014). Therefore, introducing competitions that involve cooperation among students into summer camps is desirable. In our summer camp, we developed cooperative robot competitions that feature age-appropriate themes. During the preparation of the summer camp, we involved a wide variety of students, including graduate students, undergraduate students, and high school students. In this way, we ensured that the content of the summer camp was fit for the targeted student population without losing its rigor.

Regarding the second challenge, as there are instructions and tutorials on various topics during the summer camps, such as robot design and programming, it would be highly desirable to evaluate how students understand the concepts and master the skills taught. Traditional tests are not the best approach, as students usually view

them negatively as a burden. In our summer camp, we developed a trivia game system to assess student learning in a highly entertaining and interactive fashion. The creation of the trivia game system is also part of the hands-on projects in the camp.

The article is organized as follows. First, we introduce the mobile robots and the jungle expedition-themed robot competition. Second, we describe the trivia game system that motivates and assesses student learning. Third, we give a detailed report of the preparation and running of the three-day summer camp. The article is concluded with some highlights of the summer camp and insights of what we learned from it.

Robots and the jungle expedition competition

Robots

This summer camp was developed based on two small mobile robots from Adept Inc.: an arm robot and a Mars Rover robot. As shown in Fig. 1, the arm robot has tracks and is equipped with an arm that has

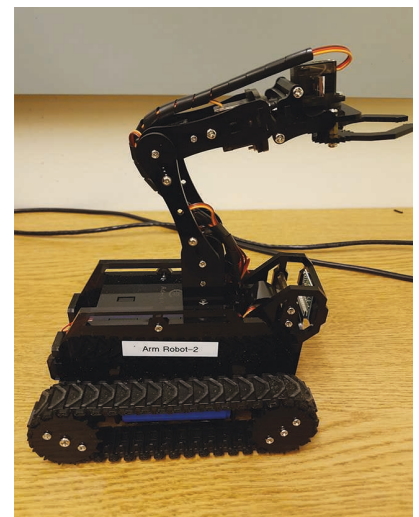


FIG1 The arm robot.

This summer camp was developed based on two small mobile robots from Adeep Inc.: an arm robot and a Mars Rover robot.

4 degrees of freedom driven by four servo motors for grabbing objects. It is powered by a Raspberry Pi 4B and equipped with a Pi camera for video streaming. The robot is controlled by a web-based interface provided by the manufacturer.

As shown in Fig. 2, the Mars Rover robot is a car-like robot powered by a Raspberry Pi 4B minicomputer and is suitable for students in grades 8–12 to learn robot building, sensor interfacing, motor control, and network programming. The front wheels are controlled by a servo motor for turning, and the rear wheels are controlled by a dc motor for driving. The robot head has

two sensors, an ultrasonic sensor and a Pi camera, through which the robot can measure the distance to obstacles and see the environment. Two servo motors are utilized to control the head pitch and yaw. Microphones, headlights, LED strips, and line-tracking sensors are also used in the robot. Assembling the robot from scratch takes approximately 5 h. A live stream from the onboard camera is displayed on a computer monitor so students can control the Mars Rover remotely through a game pad. In the summer camp, the students were required to build the Mars Rover robot only, not the arm robot. The arm robot was preassembled

and shared by all teams in the robot competition.

The Mars Rover is controlled using a client and server architecture. A separate Raspberry Pi 4B serves as a client where a joystick is connected. The Raspberry Pi 4B on the Mars Rover robot is the server that controls all onboard modules. The server executes the commands received from the client and sends the sensor data and live stream back to the client. Based on the code provided by Adeep Inc., we developed the client and server communication for students to control the robot through a game pad, which is more user friendly than the original web-based control interface provided by the company. In addition, for the client, we customized the existing interface to make it easier for students to use.

Figure 3 shows the software architecture for Mars Rover control. We call the client *ASCC_Client*. In *ASCC_Client*, the data obtained from the joystick are sent to the server. The control events include robot head movement, speed and turn, the LED light, steering, and photo taking. The live-stream image, distance, robot action status, and steering status are displayed on the user interface. The server is called *ASCC_Server*, which receives the commands sent by *ASCC_Client* and executes them. *ASCC_Server* also provides the sensor status feedback to the client.

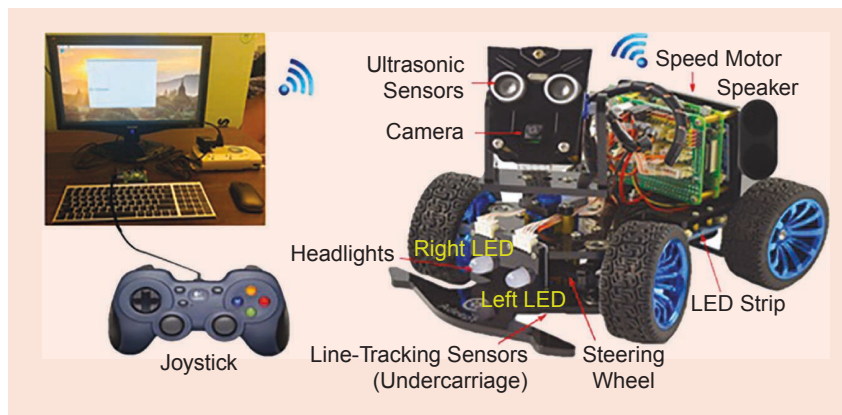


FIG2 The Mars Rover robot and its control interface.

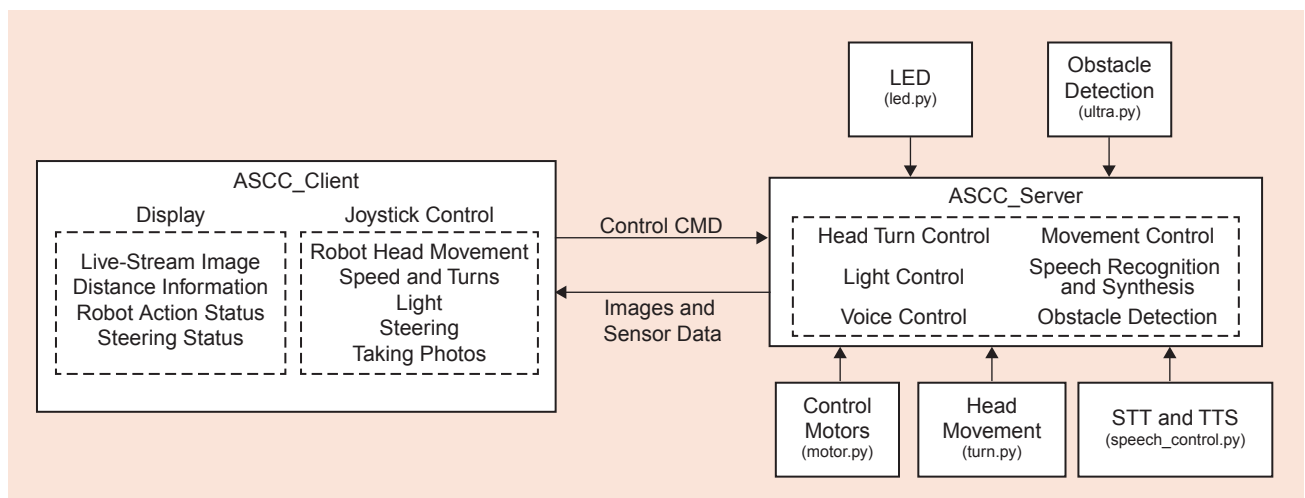


FIG3 The software architecture for Mars Rover control. CMD: command; STT: speech to text; TTS: text to speech.

We designed a set of exercises, shown in Table 1, to teach students step by step how to program the Mars Rover, which is.

Jungle expedition competition

Overview of the competition

The robot competition is designed with an age-appropriate theme called *jungle expedition*, which is as follows:

“Your team of research scientists has been studying ancient cultures in the Ontulan jungle for almost 15 years now. You have discovered many of the secrets of the Ontulan people many thought had long passed away. One of these secrets is

the location of an ancient golden relic that the Ontulan people thought was sacred. You will set out on an expedition to find this relic, but it won't be easy. You will need to build, program, and drive your jungle expedition vehicles to cross the Ontulan jungle. There are large caverns with many dead ends that you will travel through; bridges that cross roaring rivers; cave systems with little to no light; and, worst of all . . . quicksand. One of your expedition vehicles is equipped with a payload storage unit, and the other is equipped with a manipulator to acquire the relic. It seems the passage to the relic is blocked by rocks. After the relic is properly loaded into the storage ve-

hicle, you need come back to your campsite as soon as you can.”

Figure 4 shows the maze for the jungle expedition, which has a dimension of 18 × 18 ft. The maze walls were built using wood planks. Two thirds of the maze is painted green to serve as a rain forest, which is decorated with stuffed wild animals and green leaves. The other one third is painted gray, with cardboard on the ground, where a cave, a river, and a bridge are set up. In addition, there is a plateau with a relic surrounded by rocks. The plateau is accessed through a ramp.

As shown in Fig. 4, the robots enter the game through the entrance of the maze. A remotely controlled traffic light is placed at the entrance to signal the start and running of the game. When the light turns green, the game is started, and two contestants operate the two robots remotely by observing the live video streams sent back by both robots. As shown in Fig. 5(a), the contestant on the left controls the arm robot through the web-based interface provided by the robot manufacturer. The contestant

TABLE 1. Exercises for robot development.	
INDEX	EXERCISE
1	Set up the Secure Shell connection
2	Basic Linux commands
3	Server and client communication
4	Access the Logitech game pad
5	Send game pad data to the server
6	Interface LED headlights
7	Play sound and speech synthesis
8	Speed and turn control
9	Speech control
10	Obstacle detection using an ultrasonic sensor
11	Robot calibration
12	Robot remote control
13	Customize the remote control using a joystick

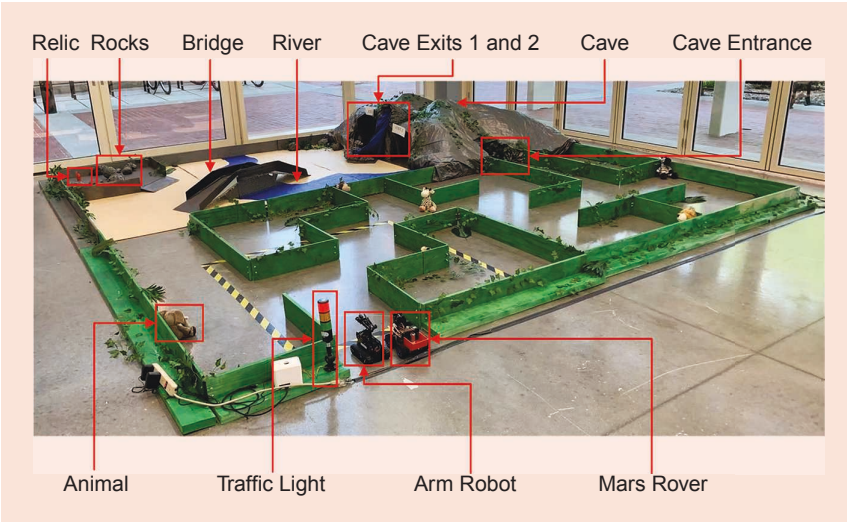


FIG4 The jungle expedition maze.



FIG5 (a) Two contestants operating the robots remotely. (b) An arm robot placing the relic into the trunk of a Mars Rover.

on the right operates the Mars Rover using the game pad through the custom interface.

The Mars Rover needs to find the animals and take their pictures on its way. The cave has one entrance and two exits. There is a 7-in LCD screen inside the cave showing which exit the robot should choose. Students are required to find the correct exit set by the referee remotely. After exiting the cave, the robots will encounter a river with a bridge. After crossing the bridge, the robots will drive up a ramp to the plateau, where the robots need to push away the rocks. Then, the arm robot will grab the relic and place it into trunk of the Mars Rover. Figure 5(b) shows a snapshot of this loading process. Figure 6 shows the



FIG6 The 3D-printed relics.

relics that we printed for the competition. After the relic is acquired, the two robots will have to return to the starting point of the maze, and the competition is finished.

Competition scoring rules

The scoring rules include five parts:

- **Maze:** The robots begin the game starting at the entrance of the maze, prompted by the traffic light. They will proceed through the maze until reaching the end at the entrance to the cave. While traveling through the maze, the robots will take pictures of the wild animals along the path, which gives them additional points. If a robot takes a wrong turn or bumps into the maze wall, it will lose points.
- **Cave:** Once in the cave, the robots need to turn on the headlights and use the camera to guide the navigation, as the cave is dark. In the corner of the cave is located a sign indicating which exit to use to leave the cave; this is randomly set through a web interface at the beginning of each

match. The robots can take a picture of the sign for an additional point. The robots should leave the cave through the correct exit. If the wrong exit is taken, the robots will encounter “quicksand,” which will result in a penalty.

- **River crossing:** Once out of the cave, the robots need to travel across the river. There is a bridge the robots can move across. Not crossing the bridge will result in a penalty.
- **Relic area:** Once on the other side of the river, the robots will ascend to the relic area, which is an enclosed area on a slightly raised platform. To access the relic, the robots will have to push the rocks away. Once cleared, the arm robot will grab the relic and place it in the truck attached to the Mars Rover vehicle. If the arm robot drops the relic, points will be deducted.
- **Return home:** Once the relic is secured, the two robots will make their way back to the entrance of the maze. Once both robots exit the maze, the game ends. Each run of the game should be completed in 8 min. If less time is used, the team will be awarded certain points.

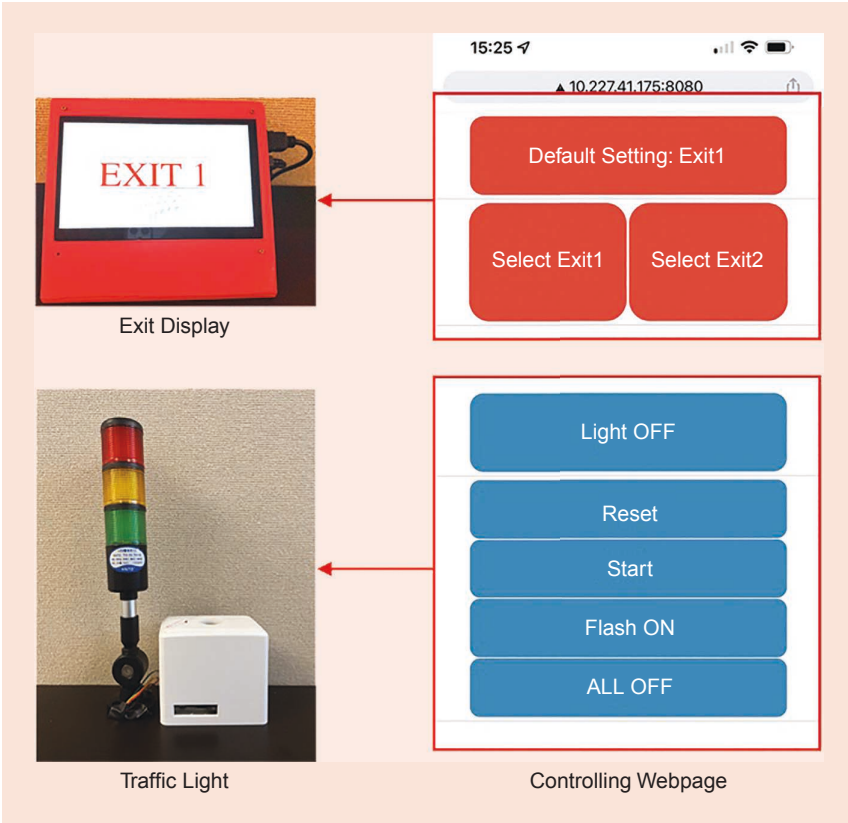


FIG7 The control interfaces for the exit display and the traffic light.

Web-based display and light control system

To provide a convenient way to control the display of the exit sign and the traffic light, we developed a web-based control system that enables the referee to randomly select an exit and control the start or stop of the competition. Figure 7 shows the web-based control interface. The referee can push the two red buttons—*Select exit 1* or *Select exit 2*—to select an exit. The blue buttons are for the traffic light. The first blue button shows the light status. The following buttons show different control commands. The web page is built using HTML and JavaScript. A Python-based web framework called *Flask* is used to develop the server, which can receive the messages sent by the client. We use TCP to send the

message from the server to the two control units: the exit display and the traffic light.

Trivia game system

The trivia game system is a tool for motivating and assessing student learning in the summer camp. As shown in Fig. 8, the system consists of a game server, multiple buzzer boxes, and an LED light showing the group number. The administrators can upload the trivia questions of the game. Then, the host can start and manage the trivia game through a smartphone. By pressing the button on the buzzer boxes, the students can buzz in and answer the questions. The scores are updated and displayed on each team's buzzer box screens. During the game, the LED light shows the binary representation of the group number of the team that buzzes in first.

Game server

The game server is the core of the system; it runs the game website and handles the events for the administrator, host, buzzer boxes, and LED light. As shown in Fig. 9, it consists of seven parts: authentication, event management, application programming interface (API), question management, team management, game management, and database. In particular, the event management part handles the events generated by the system, such as starting or resetting the games, updating the scores, buzzing in, etc. It broadcasts the events to the buzzer boxes and the LED light. The system also provides APIs for the buzzer boxes to interact with the system. The administrator enters the questions and team information through question management and team management, respectively. Moreover, the game management allows the contestants to see the questions on the contestant pages. It also allows the host to check the answers, update scores, and control the game progress. The game system is programmed using the NextJS framework,

which is a lightweight framework for web applications.

Buzzer box

The buzzer box contains two parts: the circuit board and the housing. The circuit board is based on a Raspberry Pi 3B+ board, which integrates a Wi-Fi module, 40 general purpose input-output (GPIO) pins, and four USB ports. The peripherals include an arcade-style button, an LCD screen, and a buzzer. The button, which has a built-in LED, is adopted to send the requests to the game server and make a sound when users press it. The LCD screen, driven by a 5-V power supply, can display two lines of characters in 16 columns. It shows the group name and score on the first and second lines, respectively. As it integrates an inter-integrated circuit interface module, it is convenient for users to set up and program. Eventually, all of the peripherals are integrated on the Raspberry

Pi board through the GPIO pins. The case houses all of the electrical components and breadboards, with a hole on the top to hold the button and a cutout in the front for the LCD screen.

As many students are new to Raspberry Pi and Python programming, we developed a set of exercises that led them step by step to eventually complete the coding of the client side of the trivia game system. Table 2 lists the exercises.

Description of the summer camp

Preparation for the camp

The preparation of the summer camp started in January 2022 and involved a wide variety of students, including four Ph.D. students, one master's degree student, three undergraduate students, and three high school students. The main tasks included

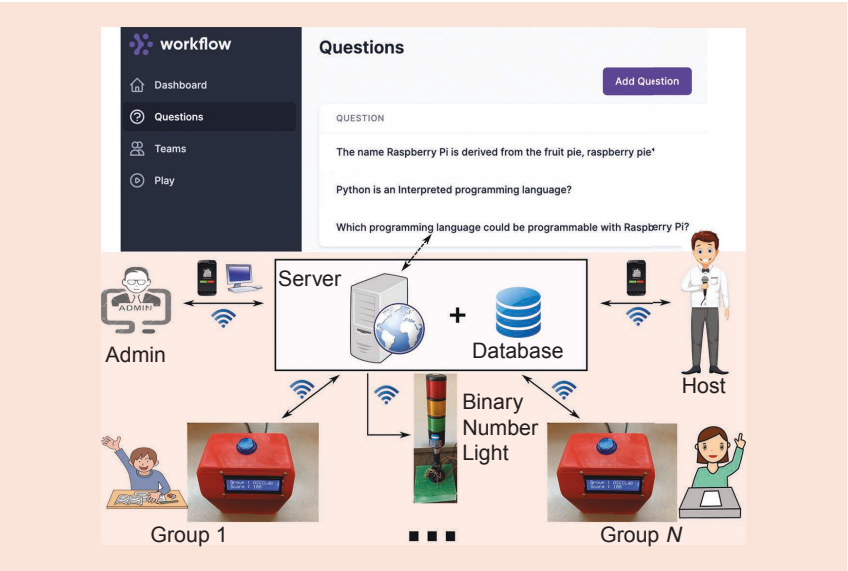


FIG8 The overall design of the trivia game system.

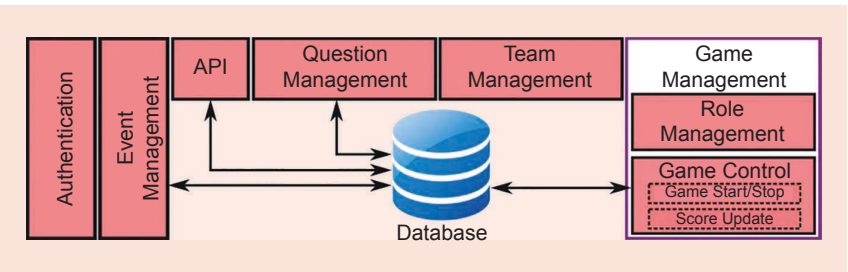


FIG9 The software architecture of the game server. API: application programming interface.

The majority of those in the beginner group did not have any prior programming experiences, and only a few of them had been exposed to simple graphical programming.

- developing the theme and competition rules
- constructing the maze for the competition
- developing the software of the robot control system
- designing the trivia game system, including both hardware and software
- developing the project handouts and tutorials
- other logistic issues, including announcements, registration, etc.

In particular, the three high school students (two 11th graders and one

12th grader) contributed to the trivia game system focusing on the game server and the game buzzer boxes. They independently implemented the codes on the server and the circuits for the LED light control. They also helped with the client-side coding of the game box, proofreading the handouts, and compiling the questions for the trivia games.

Overview of the summer camp

This summer camp was held on 22–24 June 2022 in the Endeavor

Building on the Oklahoma State University (OSU) campus. A total of 24 students (14 females and 10 males) participated in this three-day summer camp. The College of Engineering, Architecture, and Technology at OSU helped with the logistics of this camp, including announcements, registration, food, t-shirts, award prizes, a lab tour, etc. Based on their grades, the students were divided into two groups: the beginner group (grades 6–8) and advanced group (grades 9–12). There were 14 students in the beginner group and 10 students in the advanced group. The majority of those in the beginner group did not have any prior programming experiences, and only a few of them had been exposed to simple graphical programming. The advanced group had more students who had some basic programming experience, and a few students had quite extensive programming experience.

Two rooms on the first floor of the Endeavor Building were used: one for lecturing, the award ceremony, and lunches and the other for the robot/game buzzer construction and coding. In addition, a large test arena was used to set up the maze for the jungle expedition competition. To prepare the students for the robot construction and coding activities, we offered three lectures to the students: 1) an introduction to robotics, 2) mechanical principles for robotics, and 3) the basics of Python programming. The content of the lectures was later tested in the trivia game competition. During the summer camp, we organized a lab tour for the students to visit the various teaching labs in the Endeavor Building.

Camp tasks

For the beginner group, the task consists of two parts: 1) assembling the game buzzer and 2) coding for the game buzzer. The first part requires the students to follow a handout with detailed assembly instructions to build the game buzzer. The main components provided are a breadboard, a buzzer, a push button with an LED light, an LCD display, and

TABLE 2. Exercises for the trivia game system.

INDEX	EXERCISE
1	Assemble the buzzer box
2	Basic Linux commands
3	Set up the programming environment
4	Display the group name and score on the LCD screen
5	Trigger button, LED, and buzzer



(a)



(b)



(c)



(d)

FIG10 Photos from the summer camp: (a) building the Mars Rover robot, (b) building the game buzzer, (c) the jungle expedition game, and (d) the trivia game.

The preparation of the summer camp involves students at different levels, including graduate students, undergraduate students, and high school students.

wires. The second part requires the students to implement a five-exercise coding project that enables the game buzzer to communicate with the server for score calculation and display. Figure 10(b) shows two students working on the game buzzer during the summer camp.

For the advanced group, the task consists of two parts: 1) assembling the mobile robot and 2) coding for the remote control of the robot. In the first part, the students are required to follow the provided assembly instructions to build the robot from the scratch with the given components, including wheels, axles, plates, sensors, and a Raspberry Pi minicomputer. In the second part, students are required to implement a five-exercise coding project that allows the robot to follow the motion commands from the joystick. Figure 10(a) shows students working on the construction of the Mars Rover robot.

Competitions and award ceremony

Two competitions were conducted on the third day of the summer camp: the jungle expedition competition and the trivia game competition. Both groups participated in the two competitions. In this way, the beginner group students and the advanced group students enjoyed the competitions using the outcome of not only their own work but also the work of others. Parents were invited to the competitions and the award ceremony as spectators. A total of more than 20 parents attended the event on the third day of the camp.

In the jungle expedition competition, each team of two students, with one controlling the Mars Rover robot and the other controlling the arm robot, ran both robots through the maze to retrieve the relic to the home base. Each team had two runs, and the better score of the two runs was chosen as the final score of the team. The final winners were determined based on each team's final score. Figure 10(c) shows the jungle expedition game. A postcamp

survey revealed that more than 95% of the respondents indicated that they liked the jungle expedition competition very much and expressed their willingness to attend a similar event in the next year or so.

The trivia game competition consisted of a list of questions administered to seven teams, and each team had two students. Each team had a base score of 100; a correct answer earned the team two points, while an incorrect answer made the team lose one point. The question list consisted of 1) questions related to the lectures given during the summer camp and 2) random age-appropriate trivia questions. Figure 10(d) shows the trivia game. The outcome of the competition was satisfying, with more than 95% of the questions correctly answered. A postcamp survey also revealed that more than 90% of the respondents said that they mastered the knowledge delivered in the lectures.

In the award ceremony, we invited the undergraduate recruitment officer in the college to give a recruitment talk to the parents and students. Each camper received a certificate of completion of the summer camp. The winners of the competitions received a certificate of their prize, and 3D-printed trophies were given to the winners along with the prizes. There were three categories of awards: trivia game awards, jungle expedition awards, and overall awards. Both categories of awards were further divided into the beginner group and advanced group. The overall awards were given to the teams that achieved the highest combined score in both the trivia game and jungle expedition competitions.

Conclusion and Discussion

The OSU STEM Robotics Summer Camp was successfully completed. This summer camp distinguishes

itself from other traditional robot-centered summer camps in the following aspects:

- This summer camp involves both a beginner group of middle school students and an advanced group of high school students, with each focusing on a different topic while sharing their project results in the competitions.
- The summer camp adopts the Raspberry Pi minicomputer to teach Python programming, networking, and robot control and web-based programming, which proved to be effective.
- The summer camp centers on highly entertaining and engaging competitions (the jungle expedition and trivia game) that stimulate student interest and passion.
- The preparation of the summer camp involves students at different levels, including graduate students, undergraduate students, and high school students.
- The use of a trivia game system can help assess student learning outcomes and stimulate learning interest.

Through the preparation process and the three-day summer camp, we have the following insights. First, including cooperative competitions in STEM activities can greatly stimulate student passion and interest, which is consistent with the existing theory, as in Creighton and Szymkowiak (2013). Second, the theme of the competitions is very crucial to the success of the camp. Age-appropriate themes can attract students' attention and raise their interest. Our jungle expedition theme and the associated trivia game achieved that goal. Third, programming is a very important skill that students can acquire quickly as long as they have the interest and a good platform to learn with. Python is a very good starting point for students in

We hope our experience with this summer camp can help other researchers develop STEM summer camps in their particular domain of expertise.

grades 8–12 since it is a powerful programming language and is easy to learn. Fourth, assessment of learning can be done in a fun and effective way through trivia games. Our trivia game system can be extended to learning assessment in other classroom teaching settings.

Although our work has mainly achieved the objectives we had, there are several aspects that could be improved. First, as many educators are also active researchers in STEM areas, it would save them time if more advanced robotics kits with more variations were available to support different summer camps. It would also help them if standard parts were available to construct competition arenas with different themes quickly. Second, there is a need to further improve the trivia game system to make it more human-friendly and easier to set up. The current design still requires a considerable amount of time for uploading the questions and switching between different questions sets.

Overall, the STEM workforce shortage is a common problem in many countries, and there is a great need to continuously renovate the way STEM education is offered. We hope our experience with this summer camp can help other researchers develop STEM summer camps in their particular domain of expertise. As technologies are rapidly evolving, we believe more innovative approaches can be developed in the future to address this important problem.

Acknowledgment

This project is supported by National Science Foundation Grants CISE/IIS 1910933 awarded to Dr. Weihua Sheng and CBET 1849279 awarded to Dr. Yujiang Xiang. Weihua Sheng is the corresponding author.

Read more about it

- “The STEM labor force of today: Scientists, engineers, and skilled

technical workers,” Nat. Sci. Found., Alexandria, VA, USA, Jan. 2023. [Online]. Available: <https://nces.nsf.gov/pubs/nsb20212/participation-of-demographic-groups-in-stem>

- B. Diekmann and T. M. Benson-Greenwald, “Fixing STEM workforce and teacher shortages: How goal congruity can inform individuals and institutions,” *Policy Insights Behav. Brain Sci.*, vol. 5, no. 1, pp. 11–18, Feb. 2018, doi: 10.1177/2372732217747889.

- K. L. Mulvey et al., “Preparing the next generation for STEM: Adolescent profiles encompassing math and science motivation and interpersonal skills and their associations with identity and belonging,” *Youth Soc.*, vol. 5, pp. 1–24, May 2022, doi: 10.1177/0044118X221085296.

- M. J. Mohr-Schroeder et al., “Developing middle school students’ interests in stem via summer learning experiences: See blue STEM camp,” *School Sci. Math.*, vol. 114, no. 6, pp. 291–301, Oct. 2014, doi: 10.1111/ssm.12079.

- J. Balaguer Álvarez, “Introduction to robotics: Importance of a summer camp as a recruiting tool for future university students,” *IEEE Rev. Iberoam. Tecnol. Aprendizaje*, vol. 12, no. 2, pp. 71–75, May 2017, doi: 10.1109/RITA.2017.2697739.

- S. Creighton and A. Szymkowiak, “The effects of cooperative and competitive games on classroom interaction frequencies,” *Procedia Soc. Behav. Sci.*, vol. 140, pp. 155–163, Aug. 2014, in *Proc. 2nd World Conf. Psychol. Sociol. (PSYSOC)*, Brussels, Belgium, Nov. 27–29, 2013. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S187704281403328X>, doi: 10.1016/j.sbspro.2014.04.402.

About the authors

Weihua Sheng (weihua.sheng@okstate.edu) earned his B.S. and M.S. degrees in electrical engineering from Zhejiang University, China, in 1997 and 1994, and his Ph.D. degree in

electrical and computer engineering from Michigan State University in May 2002. He is a professor at the School of Electrical and Computer Engineering, Oklahoma State University (OSU), Stillwater, OK 74078 USA, and he is the director of the Laboratory for Advanced Sensing, Computation, and Control (<https://ascclab.org/>) at OSU. His research interests include social robotics, wearable computing, human robot interaction, and intelligent transportation systems. He served as an associate editor for *IEEE Robotics and Automation Magazine*. He is a Senior Member of IEEE.

Yujiang Xiang (yujiang.xiang@okstate.edu) earned his Ph.D. degree from the University of Iowa. He is an associate professor of mechanical and aerospace engineering at the School of Electrical and Computer Engineering, Oklahoma State University, Stillwater, OK 74078 USA. His research interests include human dynamics and control, human motion prediction, musculoskeletal modeling, and human robot collaboration.

Zhidong Su (zhidong.su@okstate.edu) is a Ph.D. candidate at the School of Electrical and Computer Engineering, Oklahoma State University, Stillwater, OK 74078 USA. His research interests include social robot, natural language processing, and machine learning.

Fei Liang (fei.liang@okstate.edu) is a Ph.D. candidate at the School of Electrical and Computer Engineering, Oklahoma State University, Stillwater, OK 74078 USA. His research interests include wearable computing and social robots.

Eric Wang (eric.wang401@gmail.com) is a junior at Stillwater High School, Stillwater, OK 74074 USA.

Albert Sheng (berezonehacking@gmail.com) is a junior at Stillwater High School, Stillwater, OK 74074 USA.

Charles Liu (charles.liu@ossm.edu) is a senior at the Oklahoma School of Science and Mathematics, Oklahoma City, OK 73104 USA.

