Embodied Computation: An Active-Learning Approach to Mobile Robotics Education

Laurel D. Riek, *Member, IEEE*

***Abstract*—This paper describes a newly designed upper-level undergraduate and graduate course, Autonomous Mobile Robots. The course employs active, cooperative, problem-based learning and is grounded in the fundamental computational problems in mobile robotics defined by Dudek and Jenkin. Students receive a broad survey of robotics through lectures, weekly assignments, and a final capstone project that includes a community outreach element and collaboration with artists. Students were assessed on several metrics from the ASEE literature; overall, they performed well in the course. The outreach event was also a success and was enjoyed by both the community attendees and student participants.**

***Index Terms*—Active learning, computer science education, cooperative learning, mobile robotics, outreach, robotics.**

I. Introduction

RECENT advances in low-cost, open source, mobile robotics platforms have allowed educators to easily incorporate hands-on projects into their courses. For computer science educators in particular, the ability to use a prebuilt robot such as the Willow Garage Turtlebot [1] allows instructors to focus on teaching the underlying algorithmic and software concepts of mobile robotics while still enabling a physical computing learning experience for students. Such systems also free students from spending a lot of time building and maintaining hardware.

This paper describes a newly designed upper-level undergraduate/graduate-level course in the Computer Science and Engineering Department, University of Notre Dame, Notre Dame, IN, titled Autonomous Mobile Robots. This course was closely modeled on similar courses taught at Carnegie Mellon University, Pittsburgh, PA; University of Massachusetts Lowell, Lowell; and Brown University, Providence, RI [2]–[4], and used pedagogical material from the RoboticsCourseWare.org initiative [5].

The course emphasized several instructional approaches that are favorably regarded in the engineering education literature and well suited to mobile robotics instruction [6]. First, the instructor employed active learning, with students immediately practicing concepts learned in lectures on the robots; this helps them engage with the course material. Second, the course featured cooperative learning, with students working on assignments together in small teams. Third, the instructor employed problem-based learning, where all lectures and assignments brought students back to the fundamental computational problems in mobile robotics defined by Dudek and Jenkin [7]:

1) Path planning: How can we move from one place to another while remaining in C_{free}? (C_{free} refers to free space in configuration space, or *C-space*.)
2) Localization: Given local measurements of C_{free}, how can a robot determine its state?
3) Perception and sensing: How can the robot determine which parts of the world are occupied (e.g., C_{free})?
4) Mapping: Assuming a robot knows where it is, how can it determine C_{free} ?
5) SLAM: How can a robot determine its pose and C_{free} if it knows neither?

The instructor also frequently emphasized how all of these computational problems relate to the notion of mechanistic embodiment [8], where the robot's cognition is entirely embedded in and constrained by its sensors and actuators, and its behavior is constrained by the limitations of its environment.

For student assessment, the instructor used criteria aligned with both TAC-ABET and the American Society for Engineering Education (ASEE), as described in detail in the assessment sections of this paper.

The goal of this paper is to describe the course in sufficient detail that it could be easily replicated at other colleges and universities and to communicate new ideas to the robotics education community for incorporating community outreach within capstone projects.

II. Course Description

A. Course Format

Autonomous Mobile Robots (AMR) is a three-credit Computer Science and Engineering (CSE) elective course offered during the 16-week Spring semester at Notre Dame. Up to 20 students may enroll in the class, with 10 slots reserved for graduate students and 10 for undergraduate students. While it is not necessary for students to have a background in robotics before enrolling, it is essential that they have significant programming experience. Undergraduates are required to have completed the CSE Department's two-semester *Fundamentals of Computing* course sequence; any non-CSE graduate students wishing to enroll are expected to have met similar prerequisites.

In Spring 2012, the course had nine graduate students from CSE, Aerospace and Mechanical Engineering, and Electrical Engineering, and eight undergraduate CSE majors (seven

Manuscript received May 22, 2012; accepted August 02, 2012. Date of publication November 12, 2012; date of current version January 30, 2013.

The author is with the Department of Computer Science and Engineering, University of Notre Dame, Notre Dame, IN 46556 USA (e-mail: lriek@nd.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TE.2012.2221716

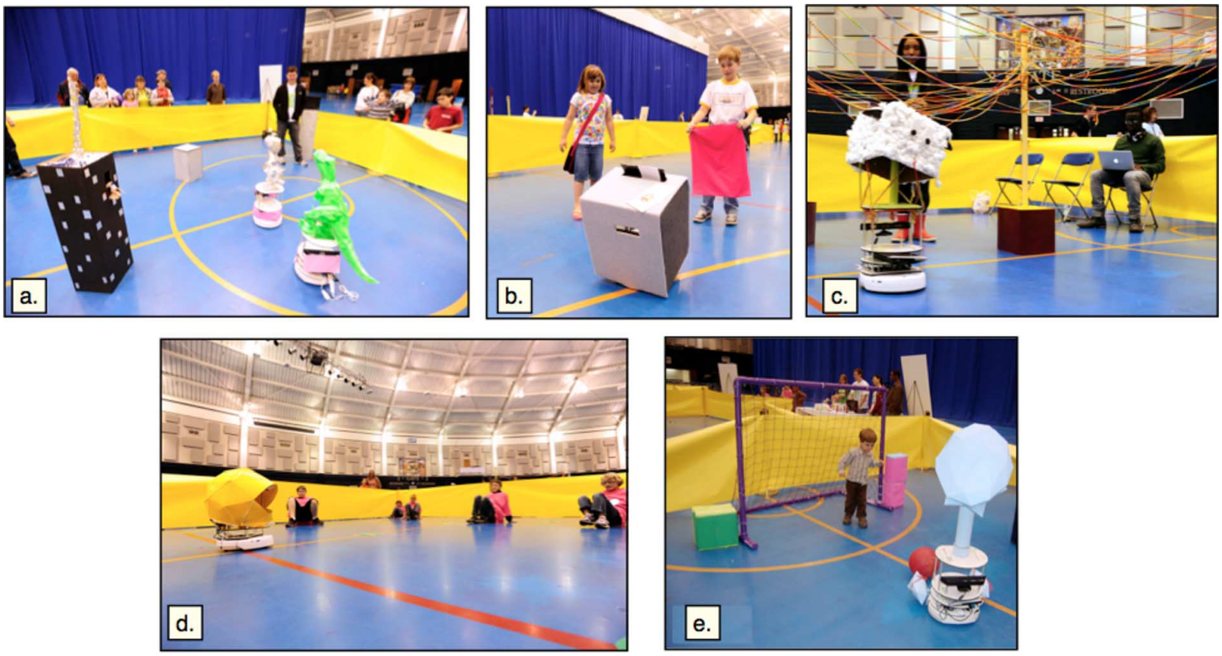

Fig. 1. Final AMR class projects at the National Robotics Week event. Project listing: (a) Robot laser tag; (b) Petbot; (c) Sheepbot (robot knitting); (d) Robot Pac-Man; (e) Robot Soccer.

seniors and one sophomore with significant programming experience).

The class has two 75-min sessions per week. The first 30–45 min of a class meeting is lecture-style, with the remaining time being given to active, cooperative learning [9], where students work in teams on assignments related to the lecture topic, with the instructor and teaching assistants available to answer questions.

Students self-select their teams and retain their teams throughout the semester. Depending on the enrollment for the semester, teams tend to contain three or four students and are typically homogeneous in composition by class (i.e., all graduate or all undergraduate).

Assignments are distributed on Tuesdays and, depending on their difficulty, are due one to two weeks later. When an assignment is due, the lab portion of class begins with a "hotwash," where teams are expected to get up in front of the classroom and informally present their approach to the previous week's lab assignment. The non-presenting students are expected to ask questions of the presenting team in order to create an interactive, cooperative learning experience for the entire class.

B. Robotic Platform and Facilities

The robotic platform students use in the course is the Turtlebot [1]. This open hardware/open software platform comprises an iRobot Create wheeled platform, a Microsoft Kinect sensor, and an ASUS laptop running Ubuntu Linux [10] and the Robot Operating System (ROS) [11]; see Fig. 1.

In Spring 2012, the course was taught in the Stinson–Remick Hall of Engineering, which provides large classrooms well suited to mobile robotics instruction. In particular, the AMR classroom provided a flexible learning space with mobile desks and computers, lockers to store the robots, and 24/7 card access to allow students to use the robots whenever they chose.

TABLE I
INSTRUCTOR'S LECTURE LIST, SPRING 2012

Class Week	Lecture Title
1	Introduction to Course and Field of Robotics; Fundamental Algorithms
2	Locomotion; Sensors
3	Sensing Algorithms
4	Representation and Planning
5	Control Architectures
6	Localization and Mapping
7	Special topic lectures related to sensing
8-14	Final Project Overview and Design; Special Topic Lectures
15	Final Project Presentations
16	Robot Ethics

C. Course Schedule

1) Weeks 1–8: During the first eight weeks of the course, students learn, in lectures, about the fundamental computational problems in mobile robotics as defined by Dudek and Jenkin [7]: sensing, representation, path planning, control, localization, and mapping. They also learn about related topics, including locomotion, kinematics, and basic optics. See Table I for a list of the instructor's lectures.

During in-class lab assignments over these weeks, students practice these concepts. The labs start with acclimating students to ROS and the Turtlebot and become progressively more complex; see Section III.

2) Weeks 8–16: During the second eight weeks of class, students learn about "special topics" in robotics, in lectures

TABLE II
GRADUATE STUDENT SPECIAL TOPIC LECTURES, SPRING 2012

Lecture Title	**Student's Area of Expertise**
Kalman Filters	Computer Vision
Robot Search and Rescue	Biometrics
Cloud Robotics	Wireless Communication
Haptic Feedback	Systems Engineering
Industrial Manipulators	Kinematics
Bipedal Robots	Bipedal Robotics
Surgical Robots	Computer Vision
Distributed Robotics and Security	Distributed Systems
Autonomous Vehicle Sensing	Computer Vision

given by the class's graduate students; this exposes the undergraduates to cutting edge research in their respective fields. In Spring 2012, the AMR graduate students' expertise across a range of robotics-related fields (including computer vision, bipedal motion, security, and distributed systems) made for interesting, well-informed lectures. See Table II for a list of the special topic lectures.

During in-class labs during this period, students focused exclusively on their final capstone project, described in detail in Section IV.

III. CURRICULUM CONTENT AND ASSESSMENT

Due to the fact that AMR is both an undergraduate and graduate course, it is designed to give students a broad survey of robotics through lectures, weekly laboratory assignments, and a final capstone project that includes a community outreach component. These are each described in more detail in this section, with assessment measures and results provided.

A. Lectures: Instructor

1) Overview: Across all lectures by the instructor, students are often reminded that robotics is a uniquely physical branch of computer science. In particular, students are made aware of how modern approaches to solving problems in robotics are very much embodied in the robot itself, intimately tied to its sensors, actuators, and environment. Many of the lecture materials for the course come from the RoboticsCourseWare.org Web site [5]. This is an ongoing effort supported by the IEEE Robotics and Automation Society to provide "an open repository for robotics pedagogical materials." As of May 2012, the Web site contains materials for 11 full robotics courses and six short robotics courses. Educators from top robotics programs around the world contributed materials, including universities such as the Massachusetts Institute of Technology (MIT); Carnegie Mellon University; École Polytechnique Fédérale de Lausanne (EPFL); National University (NU) of Singapore; University of Southern California; and the University of Pennsylvania.

2) Assessment: The primary student assessment metric for the lectures was the students' performance on the in-class lab assignments and final project. This is discussed in Sections III-C and III-D.

Qualitatively, students reported on their course evaluation forms that they enjoyed the course format of a short lecture coupled with a hands-on lab. Students completed a 5-point discrete visual analog scale (DVAS) where $\text{Strongly Agree} = 5$, and $\text{Strongly Disagree} = 1$, $M = 4.0$, $\text{s.d.} = 1.36$.

B. Lectures: Graduate Student Special Topics

1) Overview: AMR graduate students who gave special topic lectures in Spring 2012 were instructed to give a broad overview of their topic, explain clearly why it is an important problem in robotics, discuss its major computational and design challenges, describe how researchers and practitioners are addressing these challenges, and discuss recent advances in the literature or in industry.

Students were instructed to base their presentation on peer-reviewed academic publications or textbooks. They also received materials on giving research talks in computer science by Dodgson [12] and Jones [13], both of which are based on peer-reviewed presentation literature.

Finally, students received a presentation assessment rubric in advance. The rubric used for this assessment was designed by Cooney and Reid [14] to satisfy TAC-ABET student criterion 3g, "an ability to communicate effectively" [15]. The rubric assesses students using a 5-point DVAS on numerous dimensions, including: their introduction, organization, language, delivery, content, visuals, and handling of questions.

2) Assessment: Overall, students did very well on the 35-point Cooney and Reid [14] measure, $M = 32.2$, $\text{s.d.} = 2.28$. The other AMR students in the "audience" during these talks were quite engaged, asking good questions of the speakers. Furthermore, on the course evaluation given at the end of the semester, students reported enjoying the special topic lectures. They answered the question, "I enjoyed the special topic lectures," on a 5-point DVAS scale, where $\text{Strongly Agree} = 5$ and $\text{Strongly Disagree} = 1$, $M = 4.5$, $\text{s.d.} = 0.52$.

Finally, on an anecdotal note, one undergraduate student told the instructor they decided to go to graduate school in robotics after being so excited by the special topic lectures.

C. Laboratory Assignments

1) Overview: In Spring 2012, AMR students completed four graded laboratory assignments during the course, and two ungraded ones. Labs 1 and 2 were based on the Brown University CS 148 Robotics Course, taught by O. C. Jenkins [16], [17], and Lab 4 was an ROS Tutorial on SLAM [18]. All of the assignments are described in detail in Table III. Labs 1–3 and 5 were graded; Labs 0 and 4 were ungraded and completed during class.

Students could work on the assignments during in-class lab time, where the instructor and two teaching assistants were available to answer questions. Students could also work on their assignments during the week, as they had 24/7 access to the robots and classroom.

2) Assessment: The students' assignments were assessed using a rubric received given to them in advance. Typically, an assignment was graded 70% on their software implementation and 30% on their written report. Fig. 2 shows a sample grading rubric from Lab 5.

Overall, students did quite well on the graded assignments; across the four graded labs on a 100-point scale, $M = 92.04$,

TABLE III
LABORATORY ASSIGNMENTS, SPRING 2012

Assignment Title	Description	Areas of Mastery
Lab 0: Class Sundries and Getting Started with the Robots	Students learn how to use the class discussion board, the class source code repository and how to hand in assignments, and how to bring up and run teleop on the Turtlebots.	SVN, ROS, and Turtlebot teleoperation
Lab 1: Jailbreak	Students learn how to create an ROS package and implement a basic controller to randomly traverse a space (bug algorithm).	ROS, basic controllers, bump sensor processing
Lab 2: Kinect Introduction	Students are introduced to the Kinect sensor, and write a controller to seek out and drive to solid-color objects in the environment.	Kinect sensor processing, blob detection, PID controllers, ROS topic subscription, ROS publishing, ROS param usage
Lab 3: Robot Chocolate Delivery Service	Students write a controller to perform waypoint navigation to travel down a corridor, navigate obstacles, and deliver chocolate to the teaching assistants.	Planning, obstacle avoidance, localization
Lab 4: SLAM Mapping Tutorial	Students learn to calibrate the odometry and gyro sensors on the Turtlebot, and how to build a map using the SLAM algorithm.	Mapping, localization, representation, sensor calibration
Lab 5: Robot Slalom	Students write a behavior-based controller to quickly and smoothly weave through a set of orange cones.	Behavior based control, behavior arbitration (motor schemas, cooperative/competitive controllers, or potential fields)

Implementation

Topic	Questions	Percentage
Color Calibration	Did you make a color calibration file?	10%
`rosparam` usage	Does your program correctly read parameters from `rosparam`?	10%
Single object seeking	Does your robot find and drive to non-occluded objects? How close does your robot get to the objects?	20%
Transitioning between objects	Does your robot properly switch to new objects after having visited one?	15%
Robustness	Does your robot run without interruption? Does your robot get stuck?	10%
Documentation	Is your code well commented? Does it follow format specified in the syllabus? Do you "cite your sources" if you use other people's algorithms / code?	10%

Written Report

Does your one-page written report clearly describe your approach to the lab, and each team member's contribution, as specified in the syllabus?

Fig. 2. Sample grading rubric from Lab 5. Grades typically were 70% implementation and 30% written report.

mean $\text{s.d.} = 8.14$. Overall their software worked correctly, and the students successfully explained their computational approach to the assignments both in their written documentation as well as in their oral presentations (i.e., the weekly hotwashes).

D. Capstone Project

1) Overview: In the second half of the course, students worked exclusively on their final project, which brought together the fundamental concepts learned during the assignments and lectures in the first half of the class. Students designed, developed, managed, and presented the entire project from start to finish.

In addition to incorporating the various computer science and engineering concepts of the class, students were also expected to participate in a National Robotics Week (NRW) community outreach activity (see Section IV), collaborate with art students from nearby St. Mary's College, Notre Dame, IN, and learn how to prepare a formal written report and presentation.

The collaboration with St. Mary's art students came about through a previously successful collaboration between robotics researchers at Notre Dame and St. Mary's Art Department [19]. The idea of artists and roboticists working together for educational purposes was pioneered by Kim *et al.* [2] at the University of Massachusetts Lowell.

For the final project, each team had one or two St. Mary's art students assigned to their team to serve as creative professionals on the project. Their role was to help brainstorm ideas, cultivate the project, and design and create artistic ideas to complement the engineering innovations. The artists built sculptures to sit atop the robots for the NRW event.

AMR students received the following learning objectives for this project:

- how to apply concepts learned in class (e.g., sensing, localization, navigation, control) to a "real-world" project that students design, implement, and test entirely by themselves;
- how to design and manage a large-scale software and design project and create something original they can show to potential employers, graduate advisors, etc.;
- how to communicate effectively and share ideas with three groups of people: 1) fellow roboticists (e.g., classmates); 2) creative professionals with varying levels of technical expertise (e.g., art students); 3) K–12 students and members of the South Bend, IN, community who want to learn more about science, technology, engineering, and math (STEM) and how robots work.

2) Assessment: Students were assessed on five measures: a formal project proposal, creating a team poster for the NRW

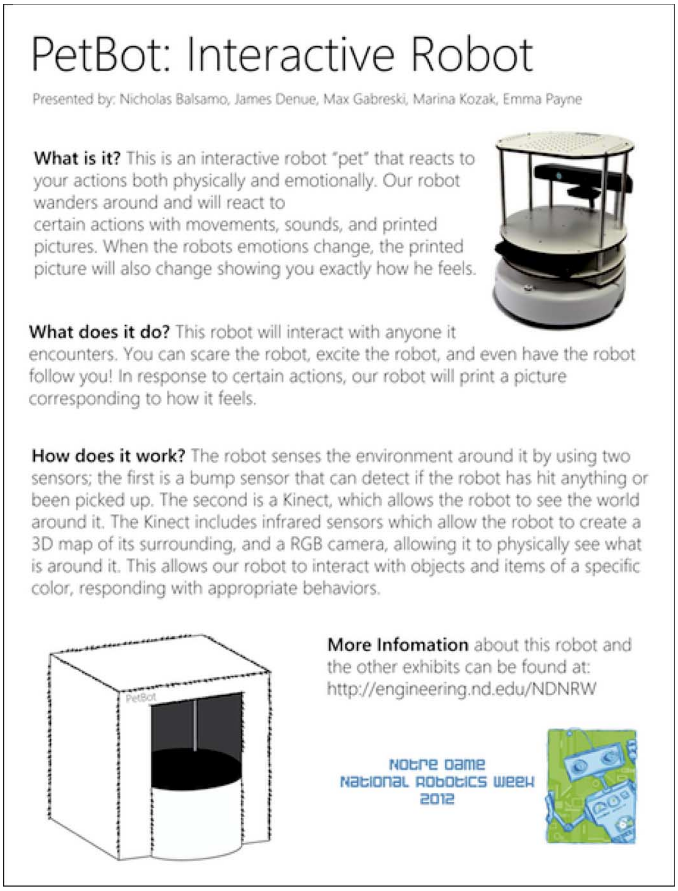

Fig. 3. Sample student-designed poster from the NRW community outreach event. Students were instructed to describe their project and how it works in a way accessible to a fifth grader.

event, participation in the NRW event, final formal presentations, and submission of a final report/software for their robot.

For the project proposal, students were required to provide a project overview, their project's software, algorithmic, and artistic components, a task list, a project schedule and Gantt chart, and a collaboration plan.

Fig. 3 shows a sample team poster that was displayed at the NRW event. Students were instructed to describe their project and describe how it works in a way accessible to a fifth grader.

Students received full credit for participating in the NRW event. They were instructed to speak to students, parents, and visitors about their project in an accessible way and were told that most attendees would not have a background in robotics or even engineering.

For the final oral presentations, students gave a 30-min talk about their project. For the presentation content, students were asked to adopt the same structure as for their final written report: Introduction, Methodology/Approach, Results, Demonstration, Discussion, and Conclusion/Future Work. They also were required to show a video or live demonstration of their robot. The oral presentation was assessed using the Cooney and Reid presentation rubric [14].

For the final written report, students were asked to prepare an 8–12-page report using the aforementioned structure. The students were assessed using the Cooney and Reid written report rubric [14].

Overall, students did very well on all assessed components of their final project. Across the six assessment items scaled to a 100-point scale, $M = 95.79$, $\text{mean s.d.} = 4.62$.

IV. National Robotics Week Outreach Activity

On April 14, 2012, the author organized a community outreach event at Notre Dame for NRW. NRW is an annual event in the US whose primary purpose is to celebrate robotics developments and educate the public about the ways in which robotics impacts society. The event is also intended to encourage K–12 students to pursue careers in STEM.

Over 65 students and faculty from Notre Dame's Colleges of Engineering and Arts and Letters, St. Mary's College, and the Robinson Community Learning Center demonstrated a variety of robotics projects. Over 600 children and adults from the South Bend community attended.

The AMR/St. Mary's project groups had five large exhibit areas in the center of the event venue for their robots, and the posters for the event were displayed prominently in front of their exhibits.

Fig. 1 shows pictures from the NRW event and children interacting with the team's final student projects. Projects included two robots that play autonomous laser tag, a robot that acts as a pet and reacts to different colors the children display, a robot called Sheepbot that autonomously knits patterns, a robot that plays a physical Pac-Man game with children who dress up as ghosts, and a robot soccer game. As previously mentioned, these projects were envisioned, designed, implemented, and created entirely by the AMR and St. Mary's students.

Generally, the students and attendees enjoyed the event. While the author and other event facilitators did not formally collect data at the event, many parents and children gave verbal feedback that they enjoyed interacting with the robots and learning about engineering. The students also reported enjoying the event, verbally and in the course evaluation. For the question, "I enjoyed participating in community outreach," students completed a 5-point DVAS scale, where *Strongly Agree* is 5 and *Strongly Disagree* is 1; $M = 4.21$, $\text{s.d.} = 0.97$.

V. Qualitative Course Evaluation

On the first day of class before the first lecture began or any course materials had been distributed, students completed a pretest that asked the following questions: Q1) "What is a robot?"; Q2) "What are robots capable of?"; and Q3) "What does it mean for a robot to be autonomous?" Students also answered these same questions in a post-test on the last day of class. Neither test was graded; they were used solely as an informal measure of the students' general understanding of robot capabilities and autonomy.

Fourteen students completed both the pretest and post-test, and several trends emerged in the data. For all three questions, all students provided more detailed answers in their post-tests compared to their pretests. Their answers included specific examples and definitions from topics discussed in class and a clear understanding of what robots are capable of, what their limitations are, and the various nuances of autonomy (e.g., mixed initiative control).

For Q1, students explicitly mentioned that robots are machines that do work for humans, operate in the physical world, and use their motors, actuators, sensors, and controllers to interact with their environment. Several students also mentioned that robots carry out goals and tasks and can have a range of autonomy.

For Q2, over half of the students said that robots are capable of doing almost anything, and are only limited by their sensors, actuators, intelligence, physics, and their developers.

For Q3, nearly half of the students added to their answer something about how robots can sense and perceive their environment, whereas only one student mentioned the environment in their pretest. This suggests the students retained the ideas of embodied computation.

VI. Conclusion

This paper presented a newly designed, upper-level undergraduate/graduate course in the Computer Science and Engineering Department of the University of Notre Dame titled Autonomous Mobile Robots. The course emphasized active learning, where students engaged in the course by practicing concepts learned in lecture immediately on robots; cooperative learning, where students worked on projects in teams; and problem-based learning, where all lectures and assignments brought students back to the fundamental computational problems in mobile robotics.

The course gave students a broad survey of robotics through the use of lectures, weekly laboratory assignments, and a final capstone project that included a community outreach component and collaboration with art students. Students were assessed on a variety of metrics from the ASEE and TAC-ABET literature and, overall, performed very well in the course. The community outreach event was a success and was an enjoyable experience both for adults and children in the community who attended as well as for the AMR students.

In future years of the course, allocating more time for the final projects would be helpful, as several teams encountered logistical problems coordinating with the artists. Furthermore, although students enjoyed working with ROS and the Turtlebots, a number of students encountered unexpected behaviors in the software and hardware that will hopefully be addressed in future releases.

Acknowledgment

The author thanks K. Hoefle, A. Huus, T. Yue, J. Birdsell, J. Brockman, J. Caponigro, K. Bowyer, P. Kilpatrick, M. D. McNally, N. Welding, D. Wordinger, and N. Gedde for their assistance with the NRW event.

References

[1] B. Gerkey and K. Conley, "Robot developer kits [ROS topics]," *IEEE Robot.Autom. Mag.*, vol. 18, no. 3, pp. 16–16, Sep. 2011.

[2] H. J. Kim, D. Coluntino, F. G. Martin, L. Silka, and H. A. Yanco, "Artbotics: Community-based collaborative art and technology education," in *Proc. ACM SIGGRAPH Educ. Program*, Aug. 2007, p. 6.

[3] J. Lalonde, C. Bartley, and I. Nourbakhsh, "Mobile robot programming in education," in *Proc. IEEE ICRA*, 2006, pp. 345–350.

[4] B. Dickinson and O. Jenkins, "Roomba Pac-Man: Teaching autonomous robotics through embodied gaming," in *Proc. AAAI*, 2007.

[5] A. Dollar, D. Rus, and P. Fiorini, "RoboticsCourseWare.org: An open repository for robotics pedagogical materials [education]," *IEEE Robot. Autom. Mag.*, vol. 15, no. 2, pp. 14–15, Jun. 2008.

[6] M. Prince, "Does active learning work? A review of the research," *J. Eng. Educ.*, vol. 93, pp. 223–231, Jul. 2004.

[7] G. Dudek and M. Jenkin, *Computational Principles of Mobile Robotics*, 2nd ed. Cambridge, U.K.: Cambridge Univ. Press, 2010.

[8] N. E. Sharkey and T. Ziemke, "Mechanistic versus phenomenal embodiment: Can robot embodiment lead to strong AI?," *Cogn. Syst. Res.*, vol. 2, no. 4, pp. 251–262, Dec. 2001.

[9] R. Felder and R. Brent, "The ABC'S of engineering education: ABET, Bloom's taxonomy, cooperative learning, and so on," in *Proc. ASEE*, 2004.

[10] Canonical Ltd., London, U.K., "About Ubuntu," 2012 [Online]. Available: http://www.ubuntu.com/project/about-ubuntu

[11] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: An open-source robot operating system," in *Proc. ICRA Workshop Open Source Softw.*, 2009.

[12] N. Dodgson, "Research skills course," 2012 [Online]. Available: http://www.cl.cam.ac.uk/teaching/1112/C00/reading/presenting.html

[13] S. P. Jones, J. Launchbury, and J. Hughes, "How to give a good research talk," in *SIGPLAN Notices*, 1993, vol. 28, no. 11, pp. 1–4.

[14] E. M. Cooney and K. Reid, "Assessment rubrics for TAC-ABET interpersonal skills," in *Proc. ASEE*, 2004.

[15] Engineering Accreditation Commission (ABET), Baltimore, MD, "Criteria for accrediting engineering programs," Tech. Rep., 2011.

[16] O. Jenkins, "CS 148 Project 1: Enclosure escape," 2011 [Online]. Available: http://www.brown-robotics.org/cs148/doku.php?id=enclosure_escape

[17] O. Jenkins, "CS 148 Project 2: Object seeking," 2011 [Online]. Available: http://www.brown-robotics.org/cs148/doku.php?id=object_seeking

[18] W. Garage, "TurtleBot odometry and gyro calibration," [Online]. Available: http://www.ros.org/wiki/turtlebot_calibration/Tutorials/Calibrate%20Odometry%20and%20Gyro

[19] T. Yue, A. E. Janiw, A. Huus, S. Aguiñaga, M. Archer, K. Hoefle, and L. D. Riek, "Creating human-robot rapport with mobile sculpture," in *Proc. 7th Annu. ACM/IEEE HRI*, Mar. 2012, p. 223.

Laurel D. Riek (M'11) is the Clare Boothe Luce Assistant Professor with the Computer Science and Engineering Department, University of Notre Dame, Notre Dame, IN. She directs the Robotics, Health, and Communication Lab, leading research on human–robot interaction, social signal processing, facial expression synthesis, and health informatics. Her research concerns new technologies to analyze and enhance communication both between people and robots, as well as between people, particularly within clinical settings. Recent work has included creating highly realistic, facially expressive medical mannequins to be used in clinical training environments, and the design of novel techniques for multimodal social sensing.