John E. Ball, M. Jean Mohammadi-Aragh,
and Christopher Goodin

Engaging Students in an Automotive Autonomy Sensor Processing Class

Incorporating active learning and high-fidelity, physics-based autonomy simulation into class projects

©SHUTTERSTOCK.COM/LUCKYSTEP

Modern signal processing (SP) classes should provide a balance between theory and application as well as use active learning exercises to engage students and facilitate learning. A new sensor processing course, Sensor Processing for Autonomous Vehicles (SPAV), was designed with two specific objectives: 1) to successfully engage students using active and collaborative learning and 2) integrate a state-of-the-art, physics-based autonomy simulator into the class.

The course was delivered to local and asynchronous distance students in spring 2020 at Mississippi State University (MSU). The MSU Autonomous Vehicle Simulator (MAVS) was used in the class. We also utilized three miniprojects to bring together theory and practice. We evaluated the course through student feedback. Results indicated that students viewed active exercises and the simulator as beneficial and useful, with multiple students describing those aspects as their favorite part of the course. Nearly all students (39 of 40) reported that they were engaged in the course.

Background

"Signal processing" is defined by the IEEE Signal Processing Society as ". . . the enabling technology for the generation, transformation, and interpretation of information" [1]. Herein, we also consider SP in a broad context, not just the traditional sampled discrete-time series data processing. For instance, deep learning (DL) image processing as well as radar and lidar object detection all come under the general SP umbrella.

Traditional digital SP (DSP) classes often are very math intensive and focus on "traditional" approaches, such as Fourier-based processing, filter structures and design, and so on. There is still a strong need to teach the fundamentals of DSP given its ubiquitous nature, yet there is also the need for classes to expose students to modern data-driven methods; current research trends; industry challenges; and opportunities in diverse spaces, such as image processing, time series processing, and lidar point cloud processing, among others. DSP classes can be rigorous, yet they can also have application- and system-level content.

Digital Object Identifier 10.1109/MSP.2021.3054327
Date of current version: 28 April 2021

 1053-5888/21©2021IEEE

Studies show that, in general, students learn better when they actively participate in meaningful learning experiences [2]. Instructors should ask many questions [2, pp. 31, 85–86] to engage students and allow them to obtain a deeper understanding. Active learning has various forms that are useful for college classrooms, including, for example, informal group discussion, think–pair–share, and minute papers. An overview of active learning can be found in [3] and [4], with a catalog of techniques in [5] and a discussion of opportunities and challenges for active learning in computing courses in [6].

Many research efforts have shown the effectiveness of active learning [3], [7], [8] and student-centric learning [9] in engineering classes. In addition to encouraging active participation, instructors should focus on creating meaningful learning situations for students. One way to make learning meaningful is to include real-world applications in the classroom. Examples and activities based on real-world applications support learning by being memorable, sparking interest, helping students connect new information to their prior understanding, and correcting misconceptions [10].

Class development

In recent years, there has been significant research and development into autonomous vehicles (AVs). The Center for Advanced Vehicular Systems (CAVS), a research center at MSU, performs AV research and development with cameras, radars, and lidars. Camera systems are ubiquitous in automotive autonomy due to high-resolution imagery, high data rates, and low cost. However, they struggle in rain, fog, and low-light situations. Radar and lidar are active technologies and, thus, allow more robust operation in fog, rain, and snow compared to red, green, blue (RGB) cameras.

Radar and lidar are fundamentally different in the data they capture and how they are processed. After working with many students over several years at CAVS in AV processing, it was apparent students had fundamental knowledge gaps in areas such as camera calibration and radar/lidar processing and that classes that addressed these gaps were needed. MSU offers multiple classes on machine learning (ML), neural networks, visualization, (traditional) DSP, image processing, and radar. However, prior to the spring 2020 semester, there was not a comprehensive introductory class on AV sensor processing.

This article outlines a special-topics class, SPAV, that was developed and delivered to 37 on-campus and 11 distance students at MSU during the spring 2020 semester. The SPAV class was designed as a 3-h, split-level (senior/master's degree-level) course focusing on sensor processing methods for cameras, radars, and lidars. To facilitate asynchronous distance students, the class did not require any hardware or equipment other than a Windows laptop (already required for all students by the MSU College of Engineering).

The Spav class was designed as a 3-hour, split-level (senior/master's degree-level) course focusing on sensor processing methods for cameras, radars, and lidars.

The class was listed as an electrical and computer engineering class, but students from any major could enroll. The prerequisite for the class is passing a junior-level signals and systems class or instructor consent. The overarching course goal was to provide a highly interactive course with both a breadth and depth of coverage in automotive autonomy topics, including terminology, the current state of the industry, state-of-the-art processing methods, strengths and weaknesses of each sensor modality, general autonomy frameworks, and control strategies and methods.

Class objectives

Objective 1: Successfully engage students using active and collaborative learning

In the course, students were given a high-level overview of how each sensor modality operates, participated in detailed discussions of the strengths and weaknesses of each sensor modality, and discussed state-of-the-art methods for sensor performance evaluation. Examples of course concepts discussed include the following:

- A camera requires demosaicing to get color imagery, calibration to correct for imperfections, and coregistration to align its images with other sensors.
- Radars operate in all-weather conditions and are good at ranging and velocity estimation, but current-generation radars do not provide high-resolution imagery.
- Lidars provide dense and rich 3D data and are good at object localization, but they are color-blind since they operate at one wavelength.

To get students involved and enhance learning, both active and collaborative learning exercises were heavily utilized in the class. Each class session incorporated two or three active or collaborative exercises, although there was one session with a data collection exercise that lasted about 45 min. These exercises engaged students, illustrated class material, helped students learn fundamental concepts, and allowed the instructor and other students to monitor the observations and conclusions of each student or student group. Since the class also had asynchronous distance students, discussion board questions were utilized to facilitate student interactions among local and distance students as well as allow the distance students to participate in active learning exercises.

Objective 2: Utilize a state-of-the-art, physics-based autonomy simulator in the class

It is widely known that complex systems like AVs require not only real-world driving tests but also simulations to provide effective testing and cover rare edge cases [11]. Baraniuk and Padgett state that using interactive simulations provides an environment where students can explore and learn [12]. Students often relate well to visual-based simulations, especially when they can change parameters and see how the results change.

Initially, three potential simulators were examined for inclusion in the SPAV class: MAVS, a noncommercial, open source software library for simulating autonomous ground vehicles; Autonomous Navigation Virtual Environment (ANVEL); and Car Learning to Act (CARLA), an open source simulator for urban autonomy research [13]. Quantum Signal, ANVEL's developer, was purchased by Ford, and the simulator is now not available for general use. Both MAVS and CARLA provide a Python application programming interface (API) and control over weather, sensors, simulation parameters, agents (vehicles and pedestrians), and so on. However, CARLA requires Linux, and most students in the class do not have Linux machines. For these reasons, MAVS was chosen to be the AV simulator and is discussed in detail in the "MAVS" section.

The primary intent of MAVS is to serve as a software library for simulating the terrain, environment, sensors, and vehicle in autonomous navigation.

Class organization and content

The class met twice a week for 75 min per meeting. The instructor advised the class to spend 1–2 h outside of class for every hour spent in the lecture. The class was organized into seven modules, each having a specific focus, and are summarized in Table 1. To assess students' progress, the class had one homework assignment for each module, three miniprojects, two exams, and a final exam.

In module 1, the students were introduced to automotive autonomy, discussed the Society of Automotive Engineering autonomy levels [14], and examined several car models to assess their autonomy levels. Module 2 gave time for the students to install the required software tools: Anaconda, Tensorflow CPU, numpy, and matplotlib for Python; MAVS; and You Only Look Once (YOLO) [15]. Several after-hours sessions were also provided to help students install the required software tools.

The fundamentals of DL were covered in module 3, including deep convolutional neural network (CNN) building blocks. This module also addressed estimating the number of parameters in each layer, which is important for embedded applications. Students also ran a CNN version of MNIST using Tensorflow and learned how to write Tensorflow code.

Module 4 covered decision, planning, and control. In this module, a proportional-integral-derivative (PID) controller was introduced, and the students examined how changing the PID controller parameters affected the response. MPC was discussed in the context of path planning. The students had an exercise where they used MPC to enable a vehicle to avoid obstacles and successfully reach the destination. Finally, reinforcement learning was introduced, and they played a simple game, stepping through the reinforcement learning system as it learned to play.

Module 5 focused on camera processing and started with a discussion of the human eye and how cameras operate in a similar manner to human rods and cones. This module then covered the basics of image demosaicing, the Bayer filter, the pinhole camera, coordinate transformation, camera calibration, stereo processing, and structure from motion. Several state-of-the-art methods were examined. Thermal and infrared (IR) cameras and how they might be used in autonomy were explained. During this module, a FLIR Systems Automotive Development Kit (ADK) (https://www.flir.com/products/adk/) long-wave IR (LWIR) camera was demonstrated to the class. The final lecture in this module was devoted to thermal imaging, and the class reviewed results from studies dealing with thermal cameras [16] to understand issues facing regular RGB cameras and how different types of thermal cameras can help in poor weather conditions.

Module 6 focused on radar processing and included topics on radar terminology, waveforms, and the radar range equation. Next, frequency-modulated continuous-wave (FMCW) radar signaling and processing were covered, including range estimation, range resolution, and maximum range calculations. The class then discussed the specifications of several automotive radars. Finally, Kalman filtering was covered and discussed in the context of adaptive cruise control.

Module 7 covered lidar processing. As most students had no previous experience with lidars, lidar architectures were explained, as were lidar terminology and design parameters (the number of beams, frame rate, maximum object range, and so on). Laser emitters, laser beam divergence, and laser detectors were discussed. Time-of-flight calculations, the lidar range equation, and atmospheric effects on

Table 1. A summary of the class modules and learning objectives.

Modules	Learning Objectives
1: Autonomy	Discuss and explain autonomy levels and basic autonomy modes
2: Tool Install	Install MAVS, Anaconda, and Python tools and utilize them in class
3: DL	Utilize DL to run advanced driver assistance systems (ADAS) processing algorithms Discuss and evaluate state-of-the-art processing methods for radar, lidar, and cameras
4: Control	Utilize proportional-integral-derivative and model predictive control
5–7: Camera, Lidar, and Radar	Explain the capabilities and limitations of radar, lidar, and camera systems Process and analyze results from real-world and simulated autonomy data sets Discuss and evaluate state-of-the-art processing methods for radar, lidar, and cameras Understand and implement basic processing steps for radar, lidar, and camera data Understand the strengths and weaknesses of radar, lidar, and camera ADAS processing

lidar were discussed. State-of-the-art methods in object detection, free-space mapping, and road detection were discussed. A Velodyne HDL-32 and VeloView (https://www.paraview.org/veloview/) were demonstrated to the class. Several scenes were captured using VeloView and, to illustrate that visualizing objects in lidar data is difficult, the students were asked to guess what objects they were seeing.

MAVS

MAVS is an interactive, real-time, physics-based simulator for autonomous ground vehicles [17]. MAVS uses physics-based ray tracing [18] to accurately simulate sensors like lidar and cameras in addition to realistic simulations of GPS sensors and microelectromechanical sensors, such as inertial measurement units and gyroscopes. Vehicle dynamics are simulated in MAVS using ReactPhysics3D [19]; MAVS can also be interfaced with other vehicle dynamics software, such as Chrono [20].

For students and researchers studying ML, MAVS can automatically generate semantically labeled data for training and testing ML algorithms.

MAVS is free and open source for noncommercial use. The core MAVS libraries are written in C++, and the code can be integrated via the C++ API or Python interface. MAVS is available on GitLab. (It can be downloaded from https://gitlab.com/cgoodin/msu-autonomous-vehicle-simulator.) Additionally, precompiled binaries for Windows 10 and Ubuntu 16.04 (MAVS precompiled binaries can be downloaded at http://www.cavs.msstate.edu/capabilities/mavs.php) and extensive online documentation are also available. (MAVS documentation is available at https://mavs-documentation.readthedocs.io/en/latest/.)

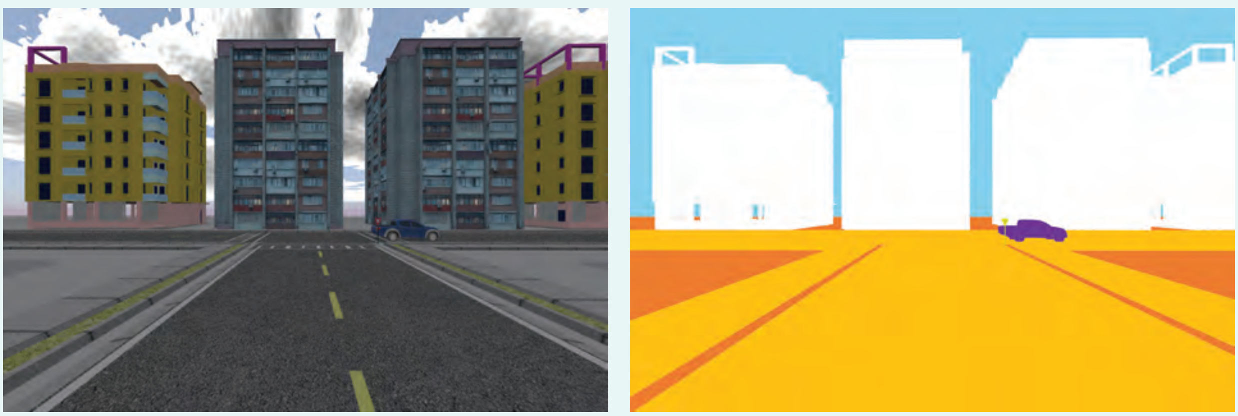
The primary intent of MAVS is to serve as a software library for simulating the terrain, environment, sensors, and vehicle in autonomous navigation. MAVS is structured to either be integrated into other applications or have other software components run in a cosimulation approach. MAVS features four basic simulation modules: vehicles, sensors, environments, and scenes.

The vehicle module provides a simulation of the vehicle motion and dynamics. The scene module defines the geometry, color, and texture of objects within the scene as well as methods for querying scene geometry using ray tracing. MAVS uses several tire and terrain interaction models to simulate driving on a variety of pavement and soil conditions and can simulate a variety of weather and environmental effects and their influences on sensor performance. The impact of rain on lidar in MAVS has been shown to match real measurements [21]. Lighting conditions based on time of day (including night) and atmospheric haziness can also be simulated with MAVS.

MAVS is being used by students, faculty, and staff at MSU to perform research in many areas of off-road autonomous operation including navigation in rough terrain, vegetation and terrain classification, negative obstacle detection, and stop sign detection. The class provided valuable distribution experience and feedback to the MAVS development team in preparation for the public release of the code (https://www.cavs.msstate.edu/capabilities/mavs.php).

For students and researchers studying ML, MAVS can automatically generate semantically labeled data for training and testing ML algorithms. The automated labeling process has been used for testing neural network-based ML algorithms for both camera [22] and lidar [23] data. Some labeled camera outputs are shown in Figure 1.

In addition to using MAVS data, students were also given databases of road scenery collected locally by the instructor and a student containing dirt and paved roads, various signage, and so on. These scenes covered highway, more country-like settings, and some urban (downtown) areas.



(a) (b)

FIGURE 1. An example of MAVS automatic semantic labeling: the (a) raw and (b) labeled image. The white label is for buildings, purple indicates vehicles, blue represents sky, yellow shows road, and orange is for ground. A yield sign can be seen labeled in light green near the truck. In particular, the extension to noisy microwave networks is discussed in detail with respect to the interface with optimization algorithms, a topic that should attract a wide readership.

Active and collaborative learning exercises

A total of 71 active and collaborative learning exercises were used in the class. Exercises included brainstorming activities, where the goal was to think of as many responses as possible; think–pair–share activities, where each student thought about a problem or question, discussed with the other students in the group, and finally came to a consensus; and various types of group exercises. Exercise lengths ranged from several minutes up to about 50 min. (The class runs for 75 min.) Most of the exercises were shorter and designed to reinforce concepts.

There were a variety of collaborative exercises involving examining and running DL or sensor processing code.

An example of a shorter exercise was listing challenges to implementing level 5 autonomy. Example student responses are edge cases, price, ethics, handling construction, working with drivers in level 0 vehicles, malfunctions, handling aggressive drivers, and so on. This exercise took about 10 min.

There were also more in-depth exercises in the class. Some examples include the following:

- having student groups take processing steps, e.g., mapping, localization, traffic prediction, and so on; explain where these tasks fit into the Eliot artificial intelligence (AI) automotive framework (a block diagram for an autonomous system); and give their rationale [24] (20 min)
- a detailed analysis of a radar Blake sheet for an FMCW radar: an Excel spreadsheet was handed out, and students investigated how certain parameters affected the radar performance) (20 min)
- a Kalman tracker simulation, where the students examined the effects of changing two parameters in the Kalman filter in a filter simulation (25 min).

In all of these cases, team results were posted to discussion boards.

Inevitably, there are gaps between theory and practice, and many algorithms or methods work well with small or limited data but might have issues in the real world. Several exercises were geared toward exploring these areas. A discussion board exercise asked students to review a paper and discuss potential difficulties encountered with using a lidar in the rain and, in a second exercise, with pedestrian detection in fog (with various types of thermal, IR, and color cameras). A different assignment asked groups of students what difficulties there could be with road detection algorithms, especially considering the many dirt and gravel roads in rural Mississippi, snow-covered scenes in northern states, and flooded areas, to name a few. Another task asked students to consider what happens in the case of a free-space mapper and path planner where there is no free space in front of the vehicle (e.g., following someone or parked in a parking lot).

There were a variety of collaborative exercises involving examining and running DL or sensor processing code. These involved groups of students and were performed in class for the local students. To facilitate asynchronous distance student involvement, the collaborative exercises culminated with the groups posting to discussion boards, where the distance students would also review and post comments in the following few days.

For example, in module 2, students examined code for a CNN to classify the MNIST digit data set. They trained the CNN and ran inference on the testing images. This exercise introduced them to DL and allowed the instructor to explain the basic MNIST CNN. A later miniproject allowed students to investigate using YOLO 9000 [15] for sign detection in simulated and real imagery.

In another instance, student groups ran two Python QT5 GUIs that demonstrated radar SP. The first GUI let them discover that, in FMCW processing, the distance of a reflecting object (we used a point target) after FMCW demodulation results in an intermediate frequency (IF) that is proportional to the object's distance. Instead of first giving them the equation that relates the IF to the object distance, the students ran simulations and hypothesized that, as the distance increases, the IF increases also. They had a visual understanding, and then we confirmed that their hypothesis is correct and that there is a linear relationship between the IF and object distance.

The second GUI gave insight into FMCW radar processing, and it allowed students to visualize automotive radar object detection. They could change the radar's FMCW parameters as well as the object's radar cross section, range, and velocity. This GUI is shown in Figure 2. The top plot on the right shows the range fast Fourier transform (FFT) results, and the bottom right plot shows the range–velocity results after velocity FFT processing. The class discussed the relationship of the IF to the object distance from the radar.

Other collaborative exercises focused on system-level information. For example, in the radar module, students used a spreadsheet and modified radar parameters for a short-range radar. When specifications were met, cells turned from red to green. Also in the radar module, students ran a Kalman filter simulation and tuned the filter parameters to see how they affected the results. In the lidar module, students examined a lidar design that calculated the maximum lidar frame rate given the field of view (FOV), number of pulses, pulse widths, and number of receivers.

In the decision, planning, and control module, students listed challenges to an autonomous system as a vehicle approaches an intersection; they also took a set of software modules defined in [25] and mapped them into the Eliot automotive framework [24]. Students were asked to explain their choices in this exercise.

Miniprojects

The classwork included three miniprojects assigned by the instructor. In each of these, local and distance students worked in teams of up to four undergraduates or four graduates (with no mixed teams). Each miniproject required the software-based assignment to be conducted. Each team submitted a report with an introduction, methodology, results, conclusions, references, and code listings. The miniprojects were designed to teach how to perform and write a small research project. Grading was based on following directions, technical content, proper IEEE citations, grammar, profession writing style, and code comment clarity. Each miniproject was worth 10% of the final grade. The timeframes were five, five, and four weeks for miniprojects 1–3, respectively.

The first miniproject allowed students to run MAVS for the first time and utilize a pretrained YOLO 9000 [15] to allow them to see how well a state-of-the-art detector would work to detect stop and yield signs in high-fidelity simulated driving imagery. Students performed experiments and wrote their results in a final report for each miniproject. Figure 3 shows example MAVS imagery.

The second miniproject was given after students had discussed camera operation and learned about camera calibration as well as camera model intrinsic and extrinsic matrices. Students collected data in class, and student groups performed an offline camera calibration procedure with the full data set and a partial data set. They then examined the calibration results and wrote a report on their findings. They discovered that you need a variety of poses and you must have samples all around the camera FOV to obtain a good calibration.

Figure 4 shows three students collecting camera calibration data in class. Distance students participated in all exercises. In the camera calibration exercise, they were not able to collect data; however, they posted their observations of how well the in-class students performed the calibration data collection, e.g., whether they got images covering a variety of the image space, different orientations of the calibration board, and so on.

Perhaps the most engaging for students was the third miniproject. Most students in the class had no experience with lidar and lidar processing. After learning about laser emitters, laser detectors, scanning lidars, and so on, they used MAVS to simulate a lidar detecting a brick on the road. The simulations examined the following lidars: Velodyne VLP-16, HDL-32E and HDL-64E; Ouster OS1 and OS2; and a Quanergy M8. The simulation estimated the number of lidar points reflected from a brick at given distances from the vehicle. The students studied how the different lidars would behave.

Class assessment: Challenges faced

There were many challenges in preparing and administering the class. Most instructors who have had to prepare a class for the first time will agree that this is a daunting task by itself. The first challenge was the depth versus breadth of the class. We wanted it to not only contain sufficient depth but also breadth as well as to focus a majority of the class on sensor processing methods for the lidar, camera, and radar sensors. To prepare students for state-of-the-art discussions, which mostly involved DL methods, an early module on DL

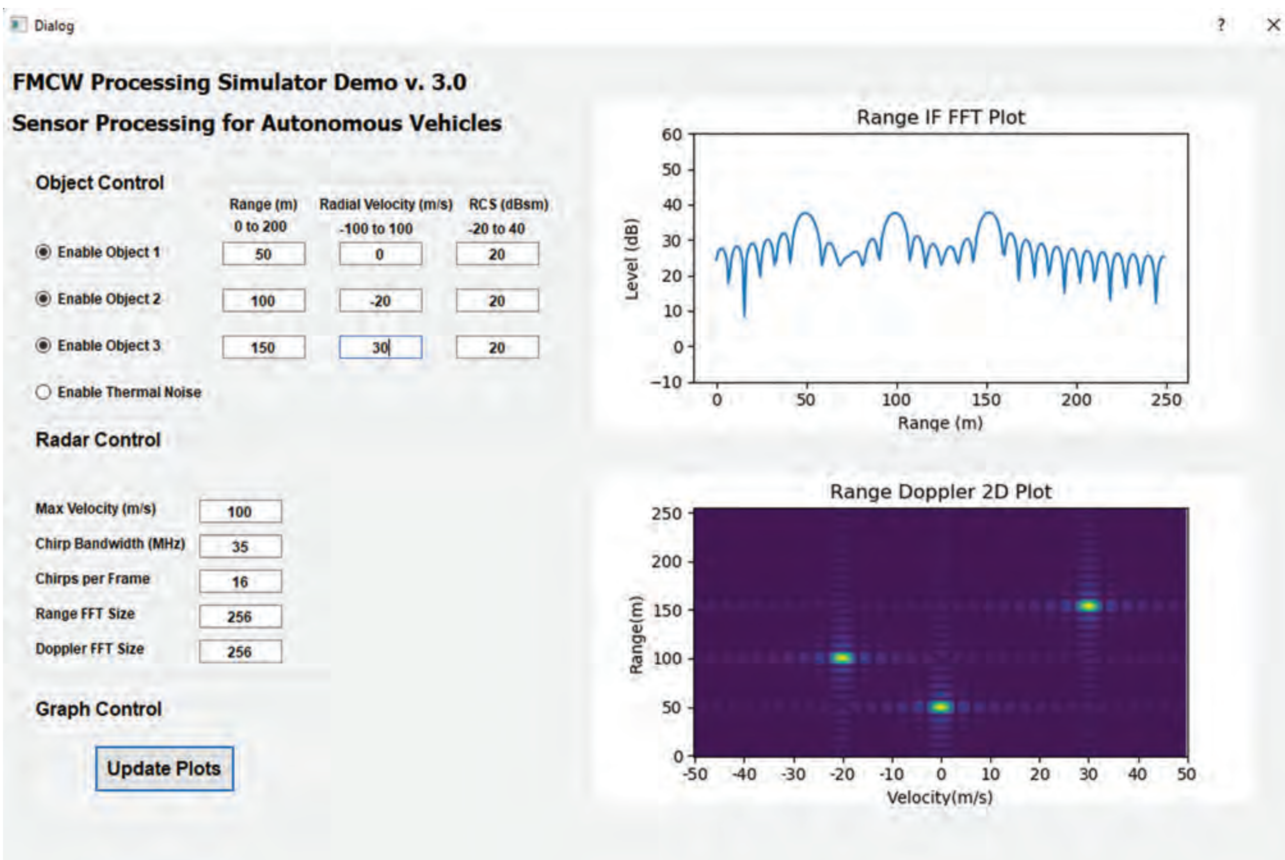


FIGURE 2. The radar FMCW processing GUI. FFT: fast Fourier transform; Max: maximum. RCS: radar cross section; IF: intermediate frequency; FFT: fast Fourier transform.

was created. Though a single class cannot cover all topics, the idea was to briefly explain major topics, such as camera intrinsic and extrinsic matrices, and expose the class to image processing topics, such as camera calibration, stereo processing, and structure from motion.

The second challenge was that there was no single book that covered the material. For a traditional DSP class, there are myriad books available. Traditional DSP is a very mature field, while automotive autonomy is changing rapidly and still in a developmental phase. Three books were selected. The first was *Creating Autonomous Vehicle Systems* [25], which covered autonomy in general; localization; perception; prediction and routing; decision, planning, and control; and reinforcement learning. To cover autonomy complexity, system framework, graceful degradation, ML, ethical issues, and so on, Eliot's book *Introduction to Driverless Self-Driving Cars: The Best of the AI Insider* [24] was chosen. *Computer Vision in Vehicle Technology: Land, Sea, and Air* was chosen to cover computer vision and vision-based autonomy systems [26].

These books did not provide adequate coverage of radar and lidar. The class materials and supplemental journal articles were used to cover these topics. We note that a very good book on autonomous radar, *Radar Signal Processing for Autonomous Driving* [27], was published too late for our course offering, but we will use the book in future classes as it is written by a nonradar expert aimed at other nonradar experts.

Covering state-of-the-art methods meant students had to read journal papers. Most graduate students are accustomed to doing this, but undergraduates are not. Having all of the students select and critique papers in a one-page writing assignment as part of each module homework gave students experience with literature reviews, how to scan a paper to find the key concepts and contributions, and how to effectively write a critique of the paper's pros and cons.

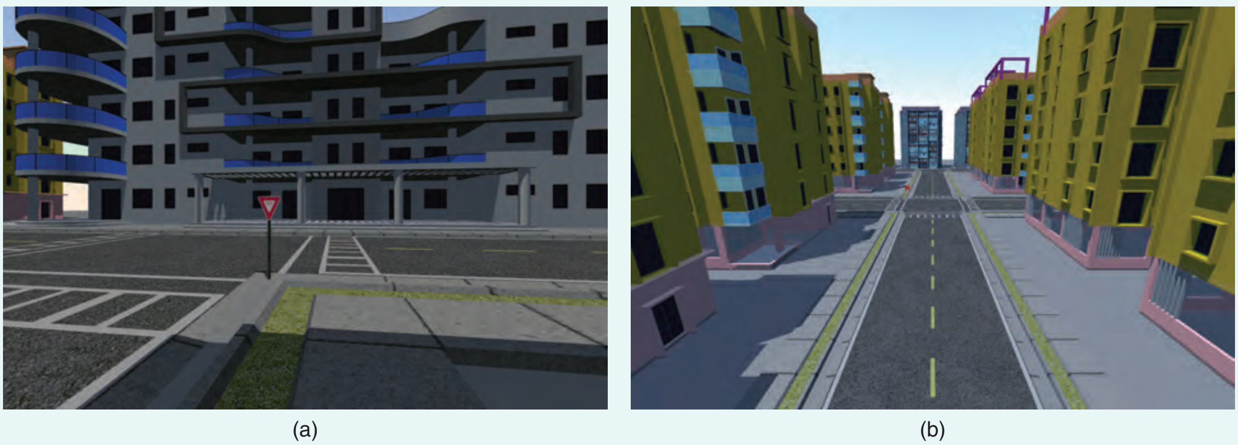


FIGURE 3. The MAVS-generated urban scenes used for class miniproject 1: (a) a yield sign and (b) a four-way intersection.

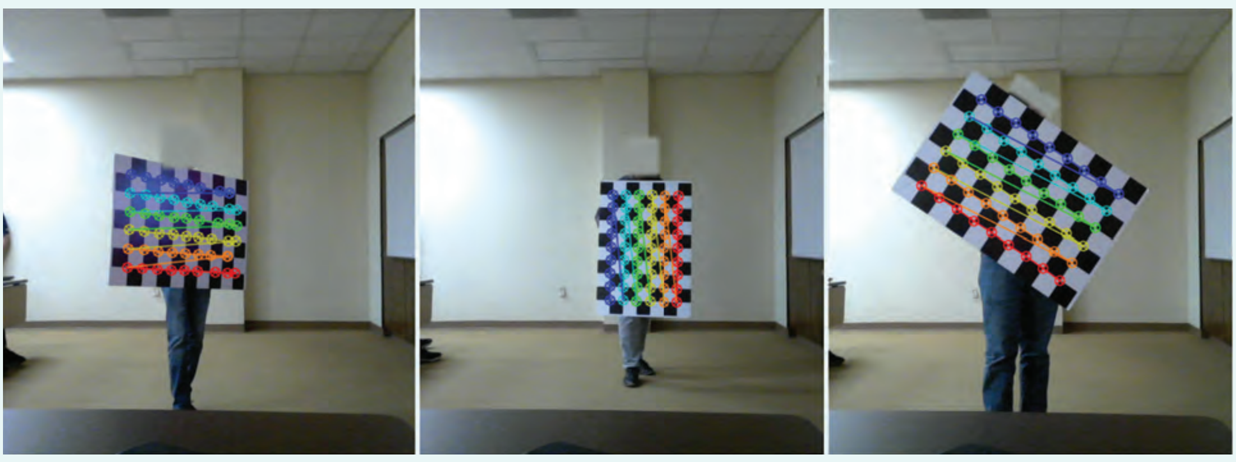


FIGURE 4. The miniproject 2 camera calibration in-class exercise.

A third challenge is the sheer amount of materials required for the class: lectures, papers for the students to read and critique, and Python codes for in-class demos. Both the DL and autonomy fields are changing rapidly, and, since there were about six class days devoted to state-of-the-art methods, these sections will need to be revised each semester as new techniques overtake the older approaches or existing methods are updated and significantly improved.

A fourth challenge was asynchronous distance teaching. Distance students often have a highly varied background, are usually working full time, and, often, have families and other duties. Most distance students work asynchronously, so they lag behind the local students since they usually watch videos at night or on weekends. Distance students also do not have the benefit of working directly with other students, unless there are several distance students who work at the same company. Keeping distance students engaged and having them feel involved is very challenging. We believed that utilizing discussion boards and having mixed groups (distance and local students) on the miniprojects helped to keep them involved.

At the conclusion of the course, numerous students described the active exercises as their favorite part of the class.

The final—and very much unexpected—challenge was the COVID-19 pandemic, which moved all MSU post-spring break classes from in person to online. Since the course materials were organized as PowerPoint presentations planned for both in-person and distance offerings, the challenges with the transition to fully online classes were somewhat mitigated. The course instructor (the first author) had never taught an online class. The distance class was taught in a special classroom and recorded for distance students to participate asynchronously. After the COVID-19 transition, a majority of the local students participated synchronously, with the active learning exercises continuing.

Several approaches changed as the class met online:

1) Before spring break, the instructor would annotate materials using the SMART board display in the distance learning classroom. For the online class, the instructor utilized a second camera and wrote on paper. Students could see the writing, and scanned versions were distributed after the class.
2) After spring break, the instructor started using WebEx Polls to poll students.
3) Most students do not prefer to interact remotely with videos on, so the instructor could not see most of the students. Before the online class, the instructor would walk around and talk to students about the exercises and provide feedback.

Student feedback

The course was designed with the two specific objectives: 1) to successfully engage students using active and collaborative learning and 2) integrate a state-of-the-art, physics-based autonomy simulator into the class. We evaluated the course through student feedback, which provided their perceptions of the course. Students provided informal feedback during the entire semester as a regular part of the active learning exercises. During the semester, there were three opportunities for students to give formal feedback to the instructor: an anonymous survey, a bonus question on the final exam, and the standard university-administered course evaluation. In this section, we discuss results from the formal survey and final exam question.

Final exam bonus question feedback

The final exam for the course included an open-ended "bonus" question that prompted, "What did you like the most about this class?" All student feedback via the bonus question was favorable. With regard to objective 1, students appreciated the active learning exercises for forcing engagement with the course topics. As an example, one student commented,

> What I liked most about this class was that you forced the class to be involved. It is easy just to sit and "attend" a lecture, but you made it fun and interactive. I also think that the class exercises were a huge help. I loved that we were able to solve the problems in class instead of only working problems at home and being lost.

Another student highlighted that the active exercises, which included both demonstrations and tinkering, helped solidify course concepts:

> My favorite part of the class was the demonstration of the various sensors and seeing them work in real time. Specifically, the in-class taking of camera calibration images, live demo of the thermal camera, and live lidar mapping of your office. To be able to visualize the output of the sensors is critical to an intuitive understanding of a sensor system. Second to that, I liked the projects that showed us the output of the camera calibration and radar display programs. Playing with parameters and seeing the effects is very satisfying.

During the lectures, students were responsive to the active learning exercises. At the conclusion of the course, numerous students described the active exercises as their favorite part of the class.

With regard to objective 2, student responses reflective positive perceptions of integrating real-world applications into the course. As expected, students highlighted how real-world applications helped them translate the theoretical course concepts to specific engineering contexts. For example, one student said,

> The combination of theory and application is what every engineering course should consist of. This class purely shows your expertise in the field, and you have the ability to hand down parts of that knowledge to us. . . . The books and articles were nice to be able to read and interpret. Getting exposure to Python, Anaconda, and MAVS are all transferable skills to take us to the next level of expertise within the field.

Another student agreed that the real-world applications enhanced the course:

I enjoyed the projects a lot. I like being able to see what we learn in class and being able to apply it in the real world. I am a very applicable and applied person, so I enjoy the applying side of the class more. Like I said, the projects allowed me to see and use what we learned in class works. I also really enjoyed learning the different ML algorithms and the image processing to track objects—that was super cool. I love how we can run these algorithms, and a computer can be trained to look at live images and pick out specific things to track or tell us is there with very high precision.

One student discussed that the real-world activities help them connect and understand fundamental SPAV course concepts with ideas learned in other undergraduate engineering courses:

My favorite part about this class was learning how to apply everything I've learned in my four years of undergraduate study. This class took everything from Python code, linear algebra, circuits, and signal processing put a major application on it, the vehicle self-driving vehicle. It has given me a lot of appreciation towards where the vehicle self-driving vehicle industry is and where it will go.

Additionally, multiple student responses specifically mentioned the benefits of using the state-of-the-art, physics-based autonomy simulator in class. They appreciated that the MAVS software was currently used to solve automotive autonomy problems—for example,

What I liked most about the class were the simulations with MAVS, demos, the Python executable codes, and the grad project. For miniproject 1, it was neat knowing that real applicable simulations could be executed with MAVS and its data could be valid to further develop approaches in automotive autonomy. The demos, such as the camera calibration, lidar point cloud analysis, and the FLIR thermal camera showcase, were very interesting.

By recalling specific aspects of the MAVS projects, such as changes to stop signs, one student indicated that the projects were memorable and achieved the goal of creating meaningful active learning:

My favorite part of the class was working on the miniprojects, especially with MAVS. The simulation of the ground vehicle was very fascinating to me, and I had a lot of fun interacting with the different variables and changing the vehicle paths and the environment variables. It was very interesting to see how slight changes to variables could greatly affect the image quality of the stop signs.

Numerous students specifically mentioned MAVS when describing their favorite aspects of the course. Several students noted that is was helpful to be able to have "hands-on" experience that applied the concepts they learned in class. It also seemed that the segmentation project using MAVS with YOLO was popular because of the visual nature of the algorithm.

The ideas of using active and collaborative learning, incorporating simulations, discussing state-of-the-art methods, and using miniprojects can be incorporated into many engineering classes.

Survey assessment

In addition to the exam question, an anonymous Qualtrics survey was administered. The questions are listed in Table 2. Of the 48 students enrolled, 40 responded to the survey, for a response rate of 83%. Overall, student responses indicate that the coverage of state-of-the-art methods and DL was beneficial (question 1). With regard to objective 1, 39 of the 40 students who responded to question 4 agreed that they felt engaged, which was a major goal for both the synchronous distance and local students. All responses indicated that students enjoyed hardware demos (question 2), and 38 of 39 students reported that the hardware demos improved their understanding of the course concepts.

Questions 6–8 in Table 2 focused on the usefulness of the active exercises (objective 1) and the simulator (objective 2) for creating a meaningful, engaged learning experience. Thirty-five of 39 students reported that the active and collaborative learning exercises were extremely or very useful for learning. When specifically asked about the MAVS software, 30 of 40 students viewed the software as useful for illustrating concepts and performing experiments (question 7), and 29 of 39 viewed MAVS as useful for learning in the general context. A few students indicated that MAVS was not useful.

In addition to the questions in Table 2, the survey included an optional open-ended question: "Briefly provide any reasoning for your views of the usefulness of visitors, student exercises, or the MAVS simulator for learning." In response to that prompt, one student noted that the active exercises and MAVS were more beneficial once students were required to connect concepts and implement automotive autonomy tasks:

The exercises and MAVS always seemed useful during the class, but became clear just how beneficial they are during the final project.

Another student perceived MAVS as useful because it allowed for students to further explore topics beyond the provided course content:

The exercises using tools and simulations really help drive the points home and to allow for experimentation with learned principles outside of class.

Students did not provide reasoning for their unfavorable ratings, but we believe negative views could be related to the specific challenges some of them encountered when using the software. While some students in the class had experience with Python programming, others did not. Additionally, during the course, a few students suggested software improvements, such as increasing the size of the simulation screen and providing more built-in file-type exports. We note that challenges like these when learning new software are not unique to MAVS.

In the course, the instructor tried to strike a balance in theory and applications. Moreover, the class was designed to fit a need for an introductory class that covered the three major autonomy sensors, their operating principles, and writing DL code to understand sensor operations and limitations. Table 2 shows that, overall, students felt that there was a relatively good balance of theory and applications. Of the 40 respondents to that question, 34 said the balance was about right, three opined that the class was a little too theory oriented, and three rated the class as too application oriented.

> "My favorite part about this class was learning how to apply everything I've learned in my four years of undergraduate study."

When designing the course, we viewed theory as a required element because we anticipated the course material would be new to most of the students. Topics such as how cameras capture data and use Bayer filters to generate RGB imagery, how camera intrinsic and extrinsic parameters are defined and how to estimate them, and how lidars and radars operate were all discussed in the class. Our initial assessment of students' prior knowledge was correct, as shown by the responses in Table 2, where 27 students indicated limited prior knowledge (a little or none). Seven students indicated they had a moderate amount of prior knowledge, with six indicating they had a lot or a great deal of prior knowledge. We believe most students' prior knowledge came from work experience at CAVS or a course on radars.

Student perceptions of the course were positive and encouraging. Through their survey responses, students reported that they were engaged in learning, enjoyed the hardware demos, and viewed the course concepts as beneficial. Students also reported that the active exercises and incorporation of the MAVS simulator in class were very or extremely useful (35 out of 39 and 29 out of 39, respectively). Students' behaviors, including class attendance, participation in activities, and posting regularly to the discussion boards, further indicated that they valued the active exercises.

Conclusions

A 3-semester-h class, SPAV, offered at the senior/master's degree level, was developed from scratch for MSU. The class focused on automotive autonomy, DL, and sensor processing for lidar, camera, and radar sensors. The class was designed to expose students to the three primary sensor systems in AVs and give them hands-on experience in sensor processing and state-of-the-art methods.

Since this class was offered as a special topics class, it will undergo another revision and offering and then be submitted to the MSU curriculum committee for adoption as a permanent class. It is the intent of the authors to strongly pursue cross-disciplinary enrollment. Currently, any engineering major can take this course as an elective. Emails advertising the new class offering will be sent to all engineering departments and researchers at CAVS so that interested students can have the opportunity to take the class.

This class was challenging to develop, and the authors do not recommend that a pretenured assistant professor undertake a new-start class that is so demanding. However, the ideas

Table 2. A summary of the survey question responses.

Question	*n*	Strongly Agree	Somewhat Agree	Neither Agree or Disagree	Somewhat Disagree	Strongly Disagree
1) The coverage of DL/modern state-of-the-art methods is very beneficial.	39	**29**	10	—	—	—
2) I enjoy the hardware demos.	40	**37**	3	—	—	—
3) The hardware demos help me understand the sensors.	39	**33**	5	1	—	—
4) I feel engaged as part of this class.	40	**33**	6	1	—	—
5) I feel challenged due to new material that is part of this class.	38	**21**	14	3	—	—
		Extremely Useful	**Very Useful**	**Moderately Useful**	**Slightly Useful**	**Not at all Useful**
6) The student exercises are beneficial (useful) for learning in the class.	39	**23**	12	3	1	—
7) The MAVS simulator is beneficial (useful) for illustrating concepts and performing experiments.	40	**25**	5	6	2	2
8) The MAVS simulator is beneficial (useful) for learning in the class.	39	**21**	8	7	1	2
		Too Theory Oriented	**A Little Too Theory Oriented**	**About Right**	**A Little Too Application Oriented**	**Too Application Oriented**
9) How well does the course balance theory and applications?	40	—	3	**34**	1	2
		A Great Deal	**A Lot**	**A Moderate Amount**	**A Little**	**None at All**
10) How much of the material in this class did you know prior to taking the class?	40	3	3	7	**21**	6

The *n* in column two indicates the total number of respondents. The largest response categories are shown in bold font.

of using active and collaborative learning, incorporating simulations, discussing state-of-the-art methods, and using mini-projects can be incorporated into many engineering classes.

Using MAVS in the class was not only beneficial to students but also valuable to the MAVS developers, as it provided a group of testers with a diverse range of experience and technical backgrounds. Students provided excellent actionable feedback for improving MAVS, pointing out the need to make the installation process easier and provide more examples and training.

The feedback on the use of active exercises and incorporation of the MAVS simulator in the class was overwhelmingly positive. Students provided informal feedback throughout the course as part of the active exercises, which was used to hone the classroom experience in real time to strengthen the learning experience.

Students also provided more formal perceptions of the course through the use of a feedback prompt and a 10-question survey. Multiple students described the active exercises or the MAVS simulator as their favorite part of the course. Student perceptions of the usefulness of the exercises and MAVS were nearly all positive. The results demonstrate that the course achieved the objectives of successfully 1) engaging students using active and collaborative learning and 2) integrating a state-of-the-art, physics-based autonomy simulator to create meaningful active learning in the classroom.

Authors

John E. Ball (jeball@ece.msstate.edu) received his Ph.D. degree in electrical engineering from Mississippi State University (MSU). After spending 16 years in several industry and government positions, he is currently an associate professor and the Robert Guyton Endowed Chair of Teaching Excellence in the Electrical and Computer Engineering Department at MSU, Mississippi State, Mississippi, 39762, USA. His research interests include sensor processing, automotive autonomy, deep learning, and effective teaching in engineering. He is a Senior Member of IEEE.

M. Jean Mohammadi-Aragh (jean@ece.msstate.edu) received her Ph.D. degree in engineering education from Virginia Tech. She is an assistant professor in the Electrical and Computer Engineering Department at Mississippi State University, Mississippi State, Mississippi, 39762, USA. Her research interests include learning analytics, pedagogical uses of digital systems, undergraduate engineering degree pathways, and diversity in engineering. She is a Member of IEEE.

Christopher Goodin (cgoodin@cavs.msstate.edu) received his Ph.D. degree in physics from Vanderbilt University. From 2008 to 2017, he worked with the U.S. Army Engineer Research and Development Center in Vicksburg, Mississippi, developing physics-based simulations of ground vehicles, sensors, and robotics. Since 2017, he has been a research faculty member with the Center for Advanced Vehicular Systems, Mississippi State University, Mississippi State, Mississippi, 39762, USA, specializing in the development of autonomous ground vehicle simulations.

References

[1] IEEE Signal Processing Society. 2020. https://signalprocessingsociety.org/ (accessed: Feb. 25, 2020).

[2] K. Bain, *What the Best College Teachers Do.* Cambridge, MA: Harvard Univ. Press, 2011.

[3] P. Baepler, J. D. Walker, D. C. Brooks, K. Saichaie, and C. I. Petersen, *A Guide to Teaching in the Active Learning Classroom: History, Research, and Practice.* Sterling, VA: Stylus, 2016.

[4] R. M. Felder and R. Brent, "Active learning: An introduction," *ASQ Higher Educ. Brief*, vol. 2, no. 4, pp. 1–5, 2009.

[5] J. L. Faust and D. R. Paulson, "Active learning in the college classroom," *J. Excell. College Teach.*, vol. 9, no. 2, pp. 3–24, 1998.

[6] K. E. Holbert and G. G. Karady, "Strategies, challenges and prospects for active learning in the computer-based classroom," *IEEE Trans. Educ.*, vol. 52, no. 1, pp. 31–38, 2009. doi: 10.1109/TE.2008.917188.

[7] M. Prince, "Does active learning work? A review of the research," *J. Eng. Educ.*, vol. 93, no. 3, pp. 223–231, 2004. doi: 10.1002/j.2168-9830.2004.tb00809.x

[8] R. Anderson, R. Anderson, K. M. Davis, N. Linnell, C. Prince, and V. Razmov, "Supporting active learning and example based instruction with classroom technology," in *Proc. SIGCSE 2007, 38th SIGCSE Techn. Symp. Comput. Sci. Educ.*, pp. 69–73. doi: 10.1145/1227310.1227338.

[9] E. Ann, J. Anthony, M. T. Kathleen, D. Janet, D. Cheryl, J. D. Filer, C. D. Wiedmaier, and C. Moore, "Students' perceptions of characteristics of effective College Teachers University of Arkansas at Fayetteville," University of Arkansas, Tech. Rep. 5, 2003.

[10] J. D. Bransford, A. L. Brown, and R. R. Cocking, *How People Learn: Brain, Mind, Experience, and School*. Washington, D.C.: National Academy Press, 2000.

[11] T. Costlow. "Complexity of autonomous-systems simulation, validation soars to the clouds." May 2019. https://www.sae.org/news/2019/06/av-simulation-testing-trends (accessed: May 17, 2020).

[12] R. G. Baraniuk and W. T. Padgett, "Trends in signal processing education [in the spotlight]," *IEEE Signal Process. Mag.*, vol. 29, no. 1, pp. 180–182, 2011. doi: 10.1109/MSP.2011.943132.

[13] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proc. 1st Annu. Conf. Robot Learn.*, 2017, pp. 1–16.

[14] *Taxonomy and Definitions for Terms Related to On-Road Motor Vehicle Automated Driving Systems*, SAE International, SAE Standard J3016_201401, 2014. [Online]. Available: https://www.sae.org/standards/content/j3016_201401/

[15] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," 2016, arXiv:1612.08242.

[16] N. Pinchon, O. Cassignol, A. Nicolas, F. Bernardin, P. Leduc, J.-P. Tarel, R. Brémond, E. Bercier et al., "All-weather vision for automotive safety: Which spectral band?" in *International Forum on Advanced Microsystems for Automotive Applications*. Berlin: Springer-Verlag, 2018, pp. 3–15.

[17] C. Goodin, M. Doude, C. R. Hudson, and D. W. Carruth, "Enabling off-road autonomous navigation-simulation of lidar in dense vegetation," *Electronics*, vol. 7, no. 9, p. 154, 2018. doi: 10.3390/electronics7090154.

[18] I. Wald, S. Woop, C. Benthin, G. S. Johnson, and M. Ernst, "Embree: A kernel framework for efficient CPU ray tracing," *ACM Trans. Graph. (TOG)*, vol. 33, no. 4, pp. 1–8, 2014. doi: 10.1145/2601097.2601199.

[19] ReactPhysics3d. https://www.reactphysics3d.com/

[20] R. Serban, M. Taylor, D. Negrut, and A. Tasora, "Chrono:: Vehicle: Template-based ground vehicle modelling and simulation," *Int. J. Veh. Perf.*, vol. 5, no. 1, pp. 18–39, 2019. doi: 10.1504/IJVP.2019.10018132.

[21] C. Goodin, D. Carruth, M. Doude, and C. Hudson, "Predicting the influence of rain on LIDAR in ADAS," *Electronics*, vol. 8, no. 1, p. 89, 2019. doi: 10.3390/electronics8010089.

[22] C. Goodin, S. Sharma, M. Doude, D. Carruth, L. Dabbiru, and C. Hudson, "Training of neural networks with automated labeling of simulated sensor data," SAE Techn. Paper 2019-01-0120, 2019.

[23] L. Dabbiru, C. Goodin, N. Scherrer, and D. Carruth, "Lidar data segmentation in off-road environment using convolutional neural networks (CNN)," SAE Techn. Paper 2020-01-0696, 2020.

[24] L. Eliot, *Introduction to Driverless Self-Driving Cars: The Best of the AI Insider.* LBE Press, 2018.

[25] S. Liu, L. Li, J. Tang, S. Wu, and J.-L. Gaudiot, *Creating Autonomous Vehicle Systems.* San Rafael, CA: Morgan & Claypool, 2017.

[26] A. M. López, A. Imiya, T. Pajdla, and J. M. Álvarez, *Computer Vision in Vehicle Technology: Land, Sea, and Air.* Hoboken, NJ: Wiley, 2017.

[27] J. Gamba, *Radar Signal Processing for Autonomous Driving.* Berlin: Springer-Verlag, 2020.

SP