INDIVIDUAL LEARNERS

UDACITY

SCHOOL OF AUTONOMOUS SYSTEMS

Sensor Fusion Engineer

Nanodegree Program Syllabus

Overview

Learn to detect obstacles in lidar point clouds through clustering and segmentation, apply thresholds and filters to radar data in order to accurately track objects, and augment the perception by projecting camera images into three dimensions and fusing these projections with other sensor data. Combine this sensor data with Kalman filters to perceive the world around a vehicle and track objects over time.

Built in collaboration with:

Mercedes-Benz

Program information

Estimated Time

4 months at 10hrs/week*

Skill Level

Advanced

*The length of this program is an estimation of total hours the average student may take to complete all required coursework, including lecture and project time. If you spend about 5-10 hours per week working through the program, you should finish within the time provided. Actual hours may vary.

Prerequisites

A well-prepared learner should have knowledge of:

- Advanced knowledge in any object-oriented programming language, preferably C++
- Intermediate probability
- Intermediate calculus
- Intermediate linear algebra
- Basic Linux command lines

Required Hardware/Software

None

Course 1

Lidar

Process raw lidar data with filtering, segmentation, and clustering to detect other vehicles on the road. Understand how to implement Ransac with planar model fitting to segment point clouds. Also implement Euclidean clustering with a KD-Tree to cluster and distinguish vehicles and obstacles.

Course Project

Lidar Obstacle Detection

Filter, segment, and cluster real point cloud data to detect obstacles in a driving environment.

Lesson 1

Introduction to Lidar & Point Clouds

- Lidar data representation.
- Working with a simulator to create PCD.
- Visualizing Lidar data.

Lesson 2

Point Cloud Segmentation

- Using PCL to segment point clouds.
- The RANSAC algorithm for planar model fitting.

Lesson 3

Clustering Obstacles

- Using PCL to cluster obstacles.
- Using a KD-Tree to store point cloud data.
- Implementing Euclidean Clustering to find clusters.
- Applying bounding boxes around clusters.

Lesson 4

Working with Real Point Cloud Data (PCD)

- Working with real self-driving car PCD data.
- Filtering PCD data.
- Playing back multiple PCD files.
- Applying point cloud processing to detect obstacles.

Course 2

Radar

Analyze radar signatures to detect and track objects. Calculate velocity and orientation by correcting for radial velocity distortions, noise, and occlusions. Apply thresholds to identify and eliminate false positives. Filter data to track moving objects over time.

Course Project

Radar Obstacle Detection

Calibrate, threshold, and filter radar data to detect obstacles in real radar data.

Lesson 1

Introduction to Radar

- Handling real radar data.
- Calculating object headings and velocities.
- Determining the appropriate sensor specifications for a task.

Lesson 2

Radar Calibration

- Correcting radar data to account for radial velocity.
- Filtering noise from real radar sensors.

Lesson 3

Radar Detection

- Thresholding radar signatures to eliminate false positives.
- Predicting the location of occluded objects.

Course 3

Camera

Fuse camera images together with lidar point cloud data. Extract object features from camera images in order to estimate object motion and orientation. Classify objects from camera images in order to apply a motion model. Project the camera image into three dimensions. Fuse this projection into three dimensions to fuse with lidar data.

Course Project

Camera & Lidar Fusion

Detect and track objects in 3D space from the benchmark KITTI dataset based on camera and lidar measurements. Compute time-to-collision based on both sensors and compare the results. Identify the best combination of keypoint detectors and descriptors for object tracking.

Lesson 1

Sensor Fusion & Autonomous Driving

- Understanding the SAE levels of autonomy.
- Comparing typical autonomous vehicle sensor sets including Tesla, Uber and Mercedes.
- Comparing camera, lidar and radar using a set of industrygrade performance criteria.

Lesson 2

Camera Technology & Collision Detection

- Understanding how light forms digital images and which properties of the camera (e.g. aperture, focal length) affect this formation.
- Manipulation images using the OpenCV computer vision library.
- Designing a collision detection system based on motion models, lidar, and camera measurements.

Lesson 3

Feature Tracking

- Detecting features from objects in a camera image using state-of-the-art detectors and standard methods.
- Matching features between images to track objects over time using state-of-the-art binary descriptors.

Lesson 4

Camera & Lidar Fusion

- Projecting 3D lidar points into a camera sensor.
- Using deep-learning to detect vehicles (and other objects) in camera images.
- Creating a three-dimensional object from lidar and camera data.

Course 4

Kalman Filters

Fuse data from multiple sources using Kalman filters. Merge data together using the prediction-update cycle of Kalman filters, which accurately track object moving along straight lines. Then build extended and unscented Kalman filters for tracking nonlinear movement.

Course Project

Unscented Kalman Filters Project

Code an Unscented Kalman Filter in C++ in order to track highly non-linear pedestrian and bicycle motion.

Lesson 1

Kalman Filters

- Constructing Kalman filters.
- Merging data from multiple sources.
- Improving tracking accuracy.
- Reducing sensor noise.

Lesson 2

Lidar & Radar Fusion with Kalman Filters

- Building a Kalman Filter in C++.
- Handling both radar and lidar data.

Lesson 3

Extended Kalman Filters

- Predicting when non-linear motion will cause errors in a Kalman filter.
- Programming an extended Kalman filter to cope with non-linear motion.
- Constructing Jacobian matrices to support EKFs.

Lesson 4

Unscented Kalman Filters

- Estimating when highly nonlinear motion might break even an extended Kalman filter.
- Creating an unscented Kalman filter to accurately track non-linear motion.

Meet your instructors.

David Silver

Curriculum Lead

David Silver leads the School of Autonomous Systems at Udacity. Before Udacity, David was a research engineer on the autonomous vehicle team at Ford. He has an MBA from Stanford and a BSE in computer science from Princeton.

Stephen Welch

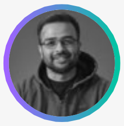
Instructor

Stephen is a content developer at Udacity and has worked on the C++ and Self-Driving Car Engineer Nanodegree programs. He started teaching and coding while completing a PhD in mathematics and has been passionate about engineering education ever since.

Andreas Haja

Professor

Andreas Haja is an engineer, educator, and autonomous vehicle enthusiast with a PhD in computer science. Andreas now works as a professor, where he focuses on project-based learning in engineering. During his career with Volkswagen and Bosch, he developed camera technology and autonomous vehicle prototypes.



Abdullah Zaidi

Instructor

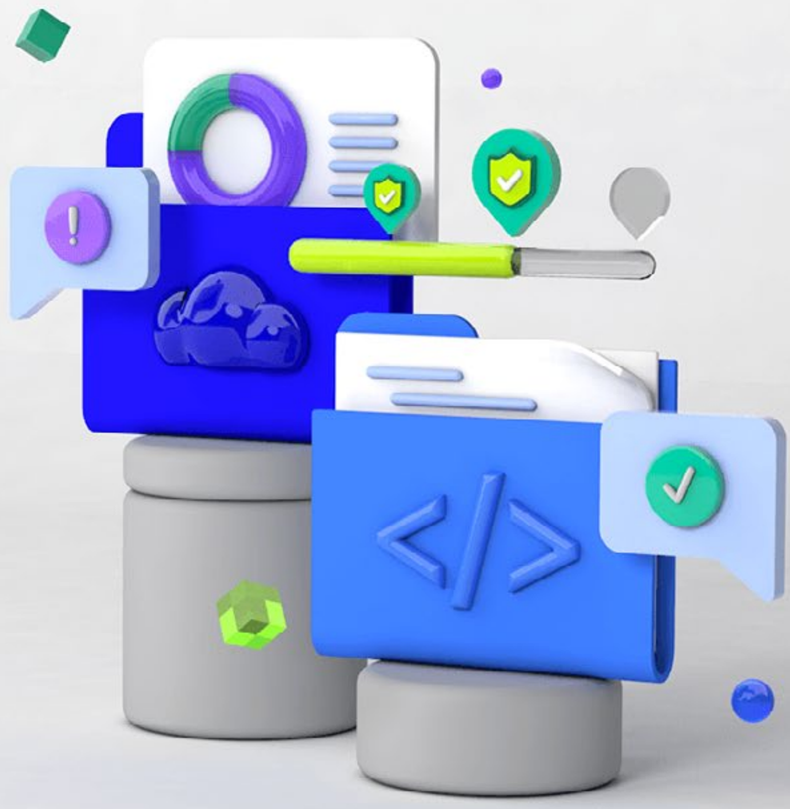
Abdullah holds his MS from the University of Maryland and is an expert in the field of radio frequency design and digital signal processing. After spending several years at Qualcomm, Abdullah joined Metawave, where he now leads radar development for autonomous driving.

Aaron Brown

Senior AV Software Engineer

Aaron Brown has a background in electrical engineering, robotics, and deep learning. Currently working with Mercedes-Benz research & development as a senior autonomous vehicle software engineer, Aaron has worked as a content developer and simulation engineer at Udacity focusing on developing projects for self-driving cars.

Udacity's learning experience



Hands-on Projects

Open-ended, experiential projects are designed to reflect actual workplace challenges. They aren't just multiple choice questions or step-by-step guides, but instead require critical thinking.

Quizzes

Auto-graded quizzes strengthen comprehension. Learners can return to lessons at any time during the course to refresh concepts.

Knowledge

Find answers to your questions with Knowledge, our proprietary wiki. Search questions asked by other students, connect with technical mentors, and discover how to solve the challenges that you encounter.

Custom Study Plans

Create a personalized study plan that fits your individual needs. Utilize this plan to keep track of movement toward your overall goal.

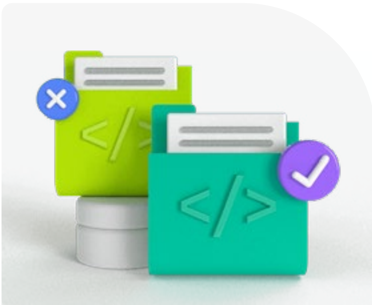
Workspaces

See your code in action. Check the output and quality of your code by running it on interactive workspaces that are integrated into the platform.

Progress Tracker

Take advantage of milestone reminders to stay on schedule and complete your program.

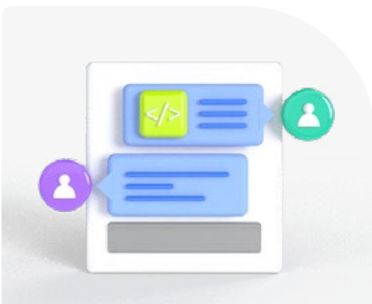
Our proven approach for building job-ready digital skills.



Experienced Project Reviewers

Verify skills mastery.

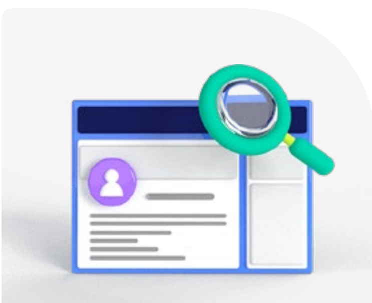
- Personalized project feedback and critique includes line-by-line code review from skilled practitioners with an average turnaround time of 1.1 hours.
- Project review cycle creates a feedback loop with multiple opportunities for improvement—until the concept is mastered.
- Project reviewers leverage industry best practices and provide pro tips.



Technical Mentor Support

24/7 support unblocks learning.

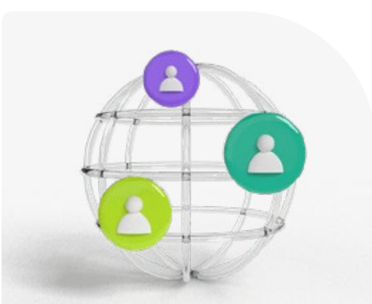
- Learning accelerates as skilled mentors identify areas of achievement and potential for growth.
- Unlimited access to mentors means help arrives when it's needed most.
- 2 hr or less average question response time assures that skills development stays on track.



Personal Career Services

Empower job-readiness.

- Access to a Github portfolio review that can give you an edge by highlighting your strengths, and demonstrating your value to employers.*
- Get help optimizing your LinkedIn and establishing your personal brand so your profile ranks higher in searches by recruiters and hiring managers.



Mentor Network

Highly vetted for effectiveness.

- Mentors must complete a 5-step hiring process to join Udacity's selective network.
- After passing an objective and situational assessment, mentors must demonstrate communication and behavioral fit for a mentorship role.
- Mentors work across more than 30 different industries and often complete a Nanodegree program themselves.

*Applies to select Nanodegree programs only.

UDACITY

Learn more at

www.udacity.com/online-learning-for-individuals →