UDACITY
FOR ENTERPRISE

THE SCHOOL OF AUTONOMOUS SYSTEMS

Self-Driving Car Engineer

NANODEGREE SYLLABUS

Overview

This Nanodegree Program is built in partnership with

Mercedes-Benz WAYMO

This program will cover the techniques that power self-driving cars across the full stack of a vehicle's autonomous capabilities. To begin, you'll learn how to apply computer vision and deep learning toward perception problems like lane finding and classifying traffic signs, as well as a full end-to-end algorithm for driving with behavioral cloning. You will also learn how to track objects from radar and lidar data with sensor fusion. From there, you'll learn and implement the concepts behind localization, path planning and control, making sure your vehicle knows where it is in the environment and how to navigate through it.

A graduate of this program will be able to:

- Train a neural network to classify images of cars, cyclists and pedestrians
- Correspond objects in camera images to objects in lidar point clouds
- Train a model to predict an object's trajectory
- Predict the behavior of other agents in the environment
- Program a finite state machine to plan a vehicle's motion in an urban environment
- Implement a Proportional Integral Derivative (PID) controller to execute a non-linear trajectory using steering, acceleration and brake

Program Information

TIME
5 months
Study 10 hours/week

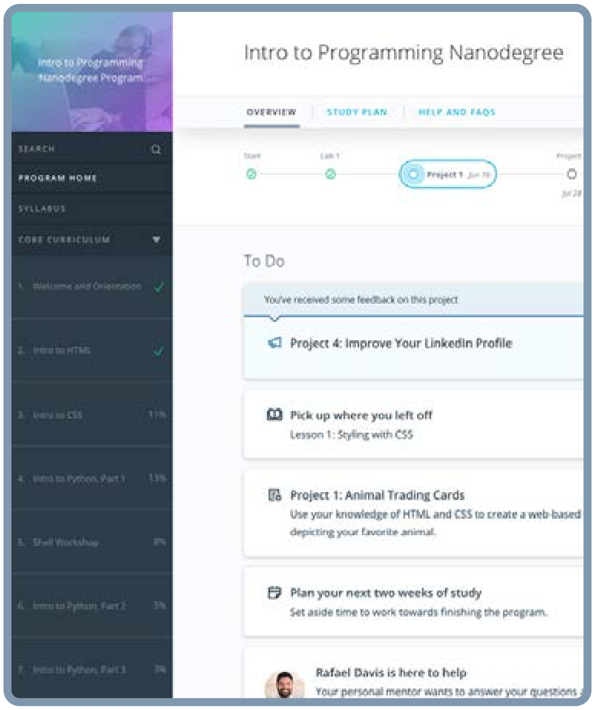
LEVEL
Specialist

PREREQUISITES
Python, C++, Linear Algebra and Calculus

HARDWARE/SOFTWARE REQUIRED
PC: Windows 7 or higher with the latest updates installed (note: Internet Explorer is not supported).
Mac: OS X 10.11 or higher with the latest updates installed.
Linux: Any recent distribution that has the supported browsers installed.
Ubuntu: 17.10+ or 14.04 LTS+.

LEARN MORE ABOUT THIS NANODEGREE
Contact us at enterpriseNDs@udacity.com.

Our Classroom Experience



REAL-WORLD PROJECTS

Learners build new skills through industry-relevant projects and receive personalized feedback from our network of 900+ project reviewers. Our simple user interface makes it easy to submit projects as often as needed and receive unlimited feedback.

KNOWLEDGE

Answers to most questions can be found with Knowledge, our proprietary wiki. Learners can search questions asked by others and discover in real-time how to solve challenges.

LEARNER HUB

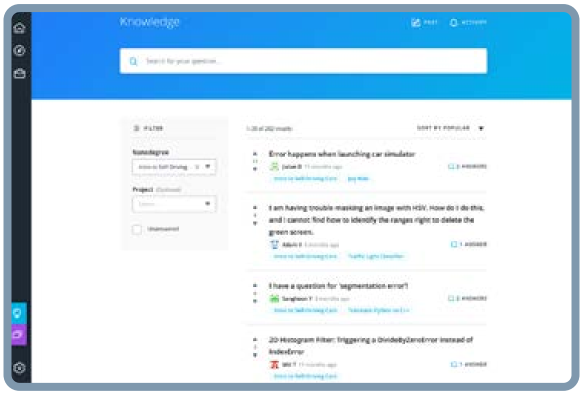
Learners leverage the power of community through a simple, yet powerful chat interface built within the classroom. Learner Hub connects learners with their technical mentor and fellow learners.

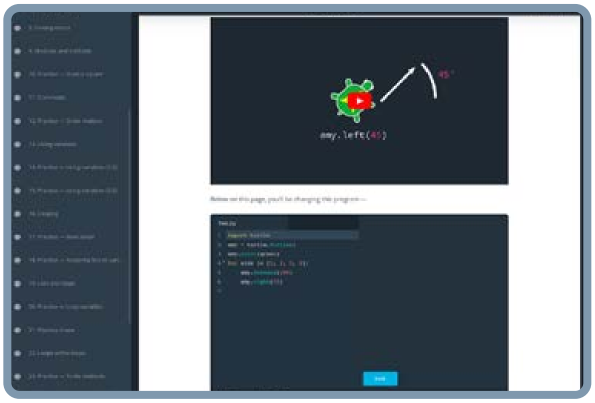
WORKSPACES

Learners can check the output and quality of their code by testing it on interactive workspaces that are integrated into the classroom.

QUIZZES

Understanding concepts learned during lessons is made simple with auto-graded quizzes. Learners can easily go back and brush up on concepts at anytime during the course.





CUSTOM STUDY PLANS

Mentors create a custom study plan tailored to learners' needs. This plan keeps track of progress toward learner goals.

PROGRESS TRACKER

Personalized milestone reminders help learners stay on track and focused as they work to complete their Nanodegree program.

Learn with the Best

Thomas Hossler

SR DEEP LEARNING ENGINEER

Thomas is originally a geophysicist but his passion for Computer Vision led him to become a Deep Learning engineer at various startups. By creating online courses, he is hoping to make education more accessible. When he is not coding, Thomas can be found in the mountains skiing or climbing.

Antje Muntzinger

SELF-DRIVING CAR ENGINEER

Antje is a self-driving car engineer and a technical lead for sensor fusion at Mercedes-Benz. She wrote her PhD about sensor fusion for advanced driver assistance systems. By educating more self-driving car engineers, she hopes to realize the dream of fully autonomous driving together in the future.

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.

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.

Learn with the Best

Munir Jojo Verge

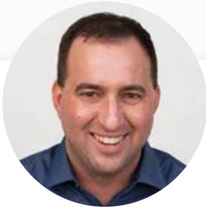
LEAD AUTONOMOUS & AI SYSTEMS DEVELOPER AT MITRE

Previously, Munir was a Motion Planning & Decision-Making Manager at Amazon. He also worked for a 2 Self-driving car companies and for WaltDisney Shanghai building TronLightcycle. Munir holds a B.Eng. in Aerospace, a M.S. in Physics and a M.S. in Space Studies.

Mathilde Badoual

FIFTH YEAR PHD STUDENT AT UC BERKELEY

Mathilde has a strong background in optimization and control, including reinforcement learning and has an engineering diploma from the electrical engineering school Supelec, in France. Previously she worked at Tesla in the energy and optimization team.



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.

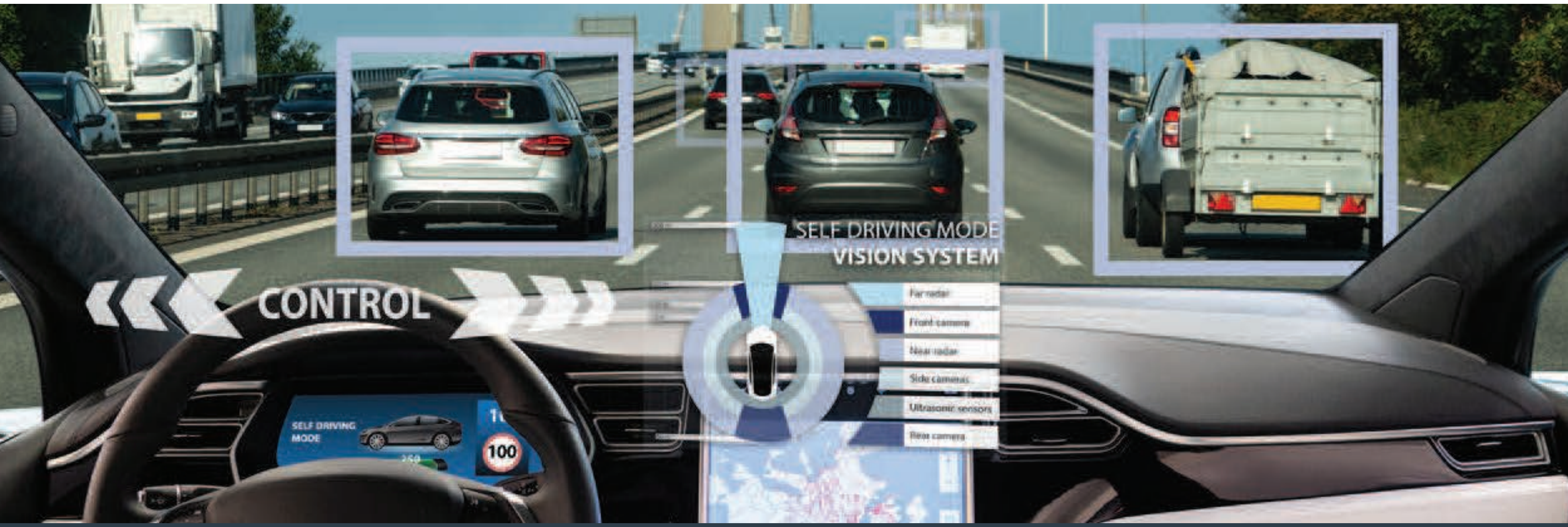
Nanodegree Program Overview

Course 1: Computer Vision

In this course, you will develop critical machine learning skills commonly leveraged in autonomous vehicle engineering. You will learn about the life cycle of a machine learning project, from framing the problem and choosing metrics to training and improving models. This course will focus on the camera sensor, and you will learn how to process raw digital images before feeding them into different algorithms, such as neural networks. You will build convolutional neural networks using TensorFlow and learn how to classify and detect objects in images. With this course, you will be exposed to the entire machine learning workflow to have a good understanding of the work of a Machine Learning Engineer and how it translates to autonomous vehicle engineering.

Project: Object Detection in an Urban Environment

In this project, students will create a convolutional neural network to detect and classify objects using data from the Waymo Open Dataset. Students are provided a dataset containing images of urban environments with annotated cyclists, pedestrians and vehicles. First, they will perform an extensive data analysis, including the computation of label distributions, displaying sample images and checking for object occlusions. This analysis will inform students to decide what augmentations are meaningful for the project. Next, they will train a neural network to detect and classify objects. Then, students will monitor the training with TensorBoard and decide when to end it. Finally, they will experiment with different hyperparameters to improve performance.



Nanodegree Program Overview

LESSON TITLE	LEARNING OUTCOMES
THE MACHINE LEARNING WORKFLOW	• Identify the key stakeholders in a ML problem • Frame the ML problem • Perform exploratory data analysis on an image dataset • Pick the most adequate model for a particular ML task • Choose the correct metric • Select and visualize the data
SENSOR AND CAMERA CALIBRATION	• Manipulate image data • Calibrate an image using checkerboard images • Perform geometric transformation of an image • Perform pixel level transformation of an image
FROM LINEAR REGRESSION TO FEEDFORWARD NEURAL NETWORKS	• Implement a logistic regression model in TensorFlow • Implement backpropagation • Implement gradient descent • Build a custom neural network for a classification task
IMAGE CLASSIFICATION WITH CONVOLUTIONAL NEURAL NETWORKS	• Write a custom classification architecture using TensorFlow • Choose the right augmentations to increase a dataset variability • Use regularization techniques to prevent overfitting • Calculate the output shape of a convolutional layer • Count the number of parameters in a convolutional network
OBJECT DETECTION IN IMAGES	• Use the TensorFlow object detection API • Choose the best object detection model for a given problem • Optimize training processes to maximize resource usage • Implement non-maximum suppression • Calculate mean average precision • Choose hyperparameters to optimize a neural network

Course 2: Sensor Fusion

In this course, you will learn about a key enabler for self-driving cars: sensor fusion. Besides cameras, self-driving cars rely on other sensors with complementary measurement principles to improve robustness and reliability. Therefore, you will learn about the lidar sensor and its role in the autonomous vehicle sensor suite. You will learn about the lidar working principle, get an overview of currently available lidar types and their differences, and look at relevant criteria for sensor selection. Also, you will learn how to detect objects such as vehicles in a 3D lidar point cloud using a deep learning approach and evaluating detection performance using a set of state-of-the-art metrics.

In the second half of the course, you will learn how to fuse camera and lidar detections and track objects over time with an Extended Kalman Filter. You will get hands-on experience with multi-target tracking, where you will learn how to initialize, update and delete tracks, assign measurements to tracks with data association techniques, managing several simultaneously. After completing the course, you will have a solid foundation to work as a sensor fusion engineer on self-driving cars.

Project: 3D Object Detection

Students will first load and preprocess 3D lidar point clouds and then use a deep learning approach to detect and classify objects (e.g., vehicles, pedestrians). They will then evaluate and visualize the objects, including calculating key performance metrics. This project combines with the Sensor Fusion project to form an entire detection pipeline.

LESSON TITLE	LEARNING OUTCOMES
INTRODUCTION TO SENSOR FUSION AND PERCEPTION	• Distinguish strengths and weaknesses of each sensor

Nanodegree Program Overview

LESSON TITLE	LEARNING OUTCOMES
THE LIDAR SENSOR	• Explain the role of lidar in autonomous driving • Extract lidar data from the Waymo dataset • Extract lidar technical properties such as coordinates • Visualize lidar data
DETECTING OBJECTS IN LIDAR	• Describe the state-of-the-art in 3D object detection • Transform a point cloud into a birds-eye view (BEV) • Perform model inference using BEV images • Visualize detection results • Evaluate object detection performance with metrics • Evaluate object detection performance between models

Nanodegree Program Overview

Project — Sensor Fusion

In this project, students will solve a challenging multi-target tracking task by fusing camera and lidar detections. They will implement an Extended Kalman filter to track several vehicles over time, including the different measurement models for camera and lidar. This task also requires a track management module for track initialization and deletion and a data association module to decide which measurement originated from which track. Finally, students will evaluate and visualize the tracked objects. To complete this project, students will use a real-world dataset, exposing them to the everyday challenges of a sensor fusion engineer.

LESSON TITLE	LEARNING OUTCOMES
KALMAN FILTERS	• Track objects over time with a linear Kalman filter
EXTENDED KALMAN FILTERS	• Track objects over time with an extended Kalman filter • Implement motion and measurement models • Derive a Jacobian for nonlinear models • Apply appropriate coordinate transforms (e.g. sensor, vehicle coordinates) • Fuse lidar measurements with camera detections with appropriate camera models
MULTI-TRACKING TRACKING	• Initialize, update and delete tracks • Define and implement a track score and track state • Calculate a simple detection probability / visibility reasoning • Associate measurements to tracks for multi-target tracking • Reduce association complexity through a gating method • Evaluate tracking performance through RMSE

Course 3: Localization

In this course, you will learn all about robotic localization, from one-dimensional motion models up to three-dimensional point cloud maps obtained from lidar sensors. You'll begin by learning about the bicycle motion model, an approach to use simple motion to estimate location at the next time step, before gathering sensor data. Next, you'll move on using Markov localization to perform 1D object tracking, as well as further leveraging motion models. From there, you will learn how to implement two different scan matching algorithms, Iterative Closest Point (ICP) and Normal Distributions Transform (NDP), working with 2D and 3D data. Finally, utilizing these scan matching algorithms in the Point Cloud Library (PCL), you will localize a simulated car with lidar sensing, using a 3D point cloud map obtained from the CARLA simulator.

Project Scan Matching Localization

In this project, students will recover the position of a simulated car using lidar with either ICP or NDT, two scan matching algorithms, aligning point cloud scans from the CARLA simulator. Students will need to achieve sufficient accuracy for the entirety of a drive within the simulated environment, updating the vehicle's location appropriately as it moves and obtains new lidar data.

LESSON TITLE	LEARNING OUTCOME
INTRODUCTION TO LOCALIZATION	• Explain how a self-driving car might use GPS or detected objects to localize itself in an environment • Predict motion to estimate location in a future time step using the bicycle motion model
MARKOV LOCALIZATION	• Apply the law of total probability to robotic motion • Derive the general Bayes/Markov filter • Implement 1D localization in C++

Nanodegree Program Overview

LESSON TITLE	LEARNING OUTCOME
CREATING SCAN MATCHING ALGORITHMS	• Explain ICP for localization • Explain NDT for localization • Implement ICP and NDT for 2D localization in C++
UTILIZING SCAN MATCHING IN 3D	• Align 3D point cloud maps with ICP • Align 3D point cloud maps with NDT • Create point cloud maps in the CARLA simulator

Course 4: Planning

Path planning routes a vehicle from one point to another, handling reactions when emergencies arise. The Mercedes-Benz Vehicle Intelligence team will take you through the three stages of path planning. First, you'll apply model-driven and data-driven approaches to predict how other vehicles on the road will behave. Then you'll construct a finite state machine to decide which one of several different maneuvers your vehicle should perform. Finally, you'll generate a safe and comfortable trajectory to execute the maneuver.

Project: Motion Planning and Decision Making for Autonomous Vehicles

In this project, you will implement two of the main components of a traditional hierarchical planner: the behavior planner and the motion planner. Both will work in unison to avoid static objects parked on the side of the road, preventing collision with these objects by executing either a "nudge" or a "lane change" maneuver, navigate intersections, and track the centerline on the traveling lane.

LESSON TITLE	LEARNING OUTCOME
BEHAVIOR PLANNING	• Learn how to think about high level behavior planning in a self-driving car
TRAJECTORY GENERATION	• Use C++ and the Eigen linear algebra library to build candidate trajectories for the vehicle to follow
MOTION PLANNING	• Program a decision making framework to plan a vehicle's motion in an urban environment • Incorporate environmental information into the motion planning algorithm • Generate an "optimal", feasible and collision free path • Navigate the vehicle through an urban driving scenario in simulation following the rules of the road, in a human-like fashion

Nanodegree Program Overview

Course 5: Control

This course will teach you how to control a car once you have the desired trajectory. In other words, how to activate the throttle and the steering wheel of the car to move it following a trajectory described by coordinates. The course will cover the most basic and most common controller: the Proportional Integral Derivative or PID controller. You will understand the basic principle of feedback controls and how they apply to autonomous driving techniques.

Project | Control and Trajectory Tracking for Autonomous Vehicles

In this project, applying the skills acquired in previous projects, you will design a PID controller to perform vehicle trajectory tracking. Given a trajectory as an array of locations and a simulation environment, you will design and code a PID controller and test its efficiency on the CARLA simulator. This project will help you understand the power and the limitations of the PID controller while utilizing feedback control. This project is good training for C++ coding, which is the standard language used in the industry.

LESSON TITLE	LEARNING OUTCOME
PID CONTROL	• Recognize the observation of the state of the vehicle (position, velocity), the action (steering, accelerator, brake) and the possible perturbations • Design and code the feedback controller (PID and MPC) for trajectory tracking, using he PID controller, and understanding how to choose the parameters to guarantee stability, and then with the MPC, a more general controller with non-linear dynamics • Test the controllers and evaluate their robustness to real-world perturbations • Analyze the differences between the two controllers

Our Nanodegree Programs Include:

Pre-Assessments

Our in-depth workforce assessments identify your team's current level of knowledge in key areas. Results are used to generate custom learning paths designed to equip your workforce with the most applicable skill sets.

Dashboard & Progress Reports

Our interactive dashboard (enterprise management console) allows administrators to manage employee onboarding, track course progress, perform bulk enrollments and more.

Industry Validation & Reviews

Learners' progress and subject knowledge is tested and validated by industry experts and leaders from our advisory board. These in-depth reviews ensure your teams have achieved competency.

Real World Hands-on Projects

Through a series of rigorous, real-world projects, your employees learn and apply new techniques, analyze results, and produce actionable insights. Project portfolios demonstrate learners' growing proficiency and subject mastery.

Our Review Process

Real-life Reviewers for Real-life Projects

Real-world projects are at the core of our Nanodegree programs because hands-on learning is the best way to master a new skill. Receiving relevant feedback from an industry expert is a critical part of that learning process, and infinitely more useful than that from peers or automated grading systems. Udacity has a network of over 900 experienced project reviewers who provide personalized and timely feedback to help all learners succeed.

Vaibhav
UDACITY LEARNER

"I never felt overwhelmed while pursuing the Nanodegree program due to the valuable support of the reviewers, and now I am more confident in converting my ideas to reality."

now at
CODING VISIONS INFOTECH

All Learners Benefit From:

- Line-by-line feedback for coding projects
- Industry tips and best practices
- Advice on additional resources to research
- Unlimited submissions and feedback loops

How it Works

Real-world projects are integrated within the classroom experience, making for a seamless review process flow.

- Go through the lessons and work on the projects that follow
- Get help from your technical mentor, if needed
- Submit your project work
- Receive personalized feedback from the reviewer
- If the submission is not satisfactory, resubmit your project
- Continue submitting and receiving feedback from the reviewer until you successfully complete your project

About our Project Reviewers

Our expert project reviewers are evaluated against the highest standards and graded based on learners' progress. Here's how they measure up to ensure your success.

900+
Expert Project Reviewers
Are hand-picked to provide detailed feedback on your project submissions.

1.8M
Projects Reviewed
Our reviewers have extensive experience in guiding learners through their course projects.

3
Hours Average Turnaround
You can resubmit your project on the same day for additional feedback.

4.85 /5
Average Reviewer Rating
Our learners love the quality of the feedback they receive from our experienced reviewers.

UDACITY
FOR ENTERPRISE

2440 W El Camino Real, #101
Mountain View, CA 94040, USA - HQ

For more information visit: **www.udacity.com/enterprise**