



NANODEGREE PROGRAM SYLLABUS

# Computer Vision Expert



# Overview

This program is designed to enhance your existing machine learning and deep learning skills with the addition of computer vision theory and programming techniques. These computer vision skills can be applied to various applications such as image and video processing, autonomous vehicle navigation, medical diagnostics, smartphone apps, and much more. This program will not prepare you for a specific career or role, rather, it will grow your deep learning and computer vision expertise, and give you the skills you need to start applying computer vision techniques to real-world challenges and applications.

The term is comprised of 3 courses and 3 projects, which are described in detail below. Building a project is one of the best ways to demonstrate the skills you've learned and each project will contribute to an impressive professional portfolio that shows potential employers your mastery of computer vision and deep learning techniques.

IN COLLABORATION WITH

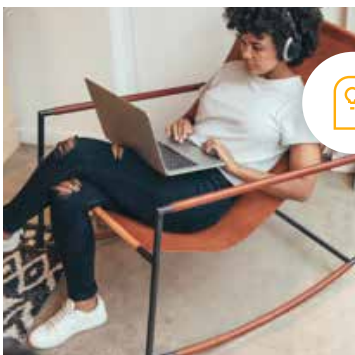
 **Affectiva**



**Estimated Time:**  
3 Months at  
10-15hrs/week



**Prerequisites:**  
Python, Statistics,  
Machine Learning,  
& Deep Learning



**Flexible Learning:**  
Self-paced, so  
you can learn on  
the schedule that  
works best for you



**Need Help?**  
[udacity.com/advisor](https://www.udacity.com/advisor)  
Discuss this program  
with an enrollment  
advisor.

# Course 1: Introduction to Computer Vision

Master computer vision and image processing essentials. Learn to extract important features from image data, and apply deep learning techniques to classification tasks.

## Course Project Facial Keypoint Detection

Use image processing techniques and deep learning techniques to detect faces in an image and find facial keypoints, such as the position of the eyes, nose, and mouth on a face.

This project tests your knowledge of image processing and feature extraction techniques that allow you to programmatically represent different facial features. You'll also use your knowledge of deep learning techniques to program a convolutional neural network to recognize facial keypoints. Facial keypoints include points around the eyes, nose, and mouth on any face and are used in many applications, from facial tracking to emotion recognition.

### LEARNING OUTCOMES

#### LESSON ONE

##### Introduction to Computer Vision

- Learn where computer vision techniques are used in industry.
- Prepare for the course ahead with a detailed topic overview.
- Start programming your own applications!

#### LESSON TWO

##### Image Representation and Analysis

- See how images are represented numerically.
- Implement image processing techniques like color and geometric transforms.
- Program your own convolutional kernel for object edge-detection.

#### LESSON THREE

##### Convolutional NN Layers

- Learn about the layers of a deep convolutional neural network: convolutional, maxpooling, and fully-connected layers.
- Build an CNN-based image classifier in PyTorch.
- Learn about layer activation and feature visualization techniques.

**LESSON FOUR****Features and Object Recognition**

- Learn why distinguishing features are important in pattern and object recognition tasks.
- Write code to extract information about an object's color and shape.
- Use features to identify areas on a face and to recognize the shape of a car or pedestrian on a road.

**LESSON FIVE****Image Segmentation**

- Implement k-means clustering to break an image up into parts.
- Find the contours and edges of multiple objects in an image.
- Learn about background subtraction for video.

# Course 2: Advanced Computer Vision and Deep Learning

Learn to apply deep learning architectures to computer vision tasks. Discover how to combine CNN and RNN networks to build an automatic image captioning application.

## Course Project Automatic Image Captioning

Combine CNN and RNN knowledge to build a deep learning model that produces captions given an input image.

Image captioning requires that you create a complex deep learning model with two components: a CNN that transforms an input image into a set of features, and an RNN that turns those features into rich, descriptive language. In this project, you will implement these cutting-edge deep learning architectures.

### LEARNING OUTCOMES

#### LESSON ONE

##### Advanced CNN Architecture

- Learn about advances in CNN architectures.
- See how region-based CNN's, like Faster R-CNN, have allowed for fast, localized object recognition in images.
- Work with a YOLO/single shot object detection system

#### LESSON TWO

##### Recurrent Neural Networks

- Learn how recurrent neural networks learn from ordered sequences of data.
- Implement an RNN for sequential text generation.
- Explore how memory can be incorporated into a deep learning model.
- Understand where RNN's are used in deep learning applications.

#### LESSON THREE

##### Attention Mechanisms

- Learn how attention allows models to focus on a specific piece of input data.
- Understand where attention is useful in natural language and computer vision applications.

#### LESSON FOUR

##### Image Captioning

- Learn how to combine CNNs and RNNs to build a complex captioning model.
- Implement an LSTM for caption generation.
- Train a model to predict captions and understand a visual scene.

# Course 3: Object Tracking and Localization

Learn how to locate an object and track it over time. These techniques are used in a variety of moving systems, such as self-driving car navigation and drone flight.

## Course Project Landmark Detection and Tracking

Use feature detection and keypoint descriptors to build a map of the environment with SLAM (simultaneous localization and mapping).

Implement a robust method for tracking an object over time, using elements of probability, motion models, and linear algebra. This project tests your knowledge of localization techniques that are widely used in autonomous vehicle navigation.

### LEARNING OUTCOMES

#### LESSON ONE

#### Object Motion and Tracking

- Learn how to programmatically track a single point over time.
- Understand motion models that define object movement over time.
- Learn how to analyze videos as sequences of individual image frames.

#### LESSON TWO

#### Optical Flow and Feature Matching

- Implement a method for tracking a set of unique features over time.
- Learn how to match features from one image frame to another.
- Track a moving car using optical flow.

#### LESSON THREE

#### Robot Localization

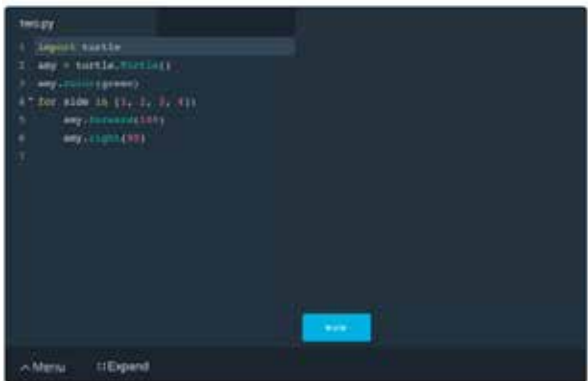
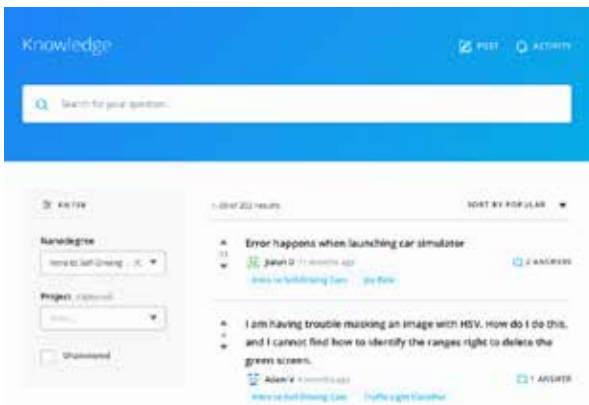
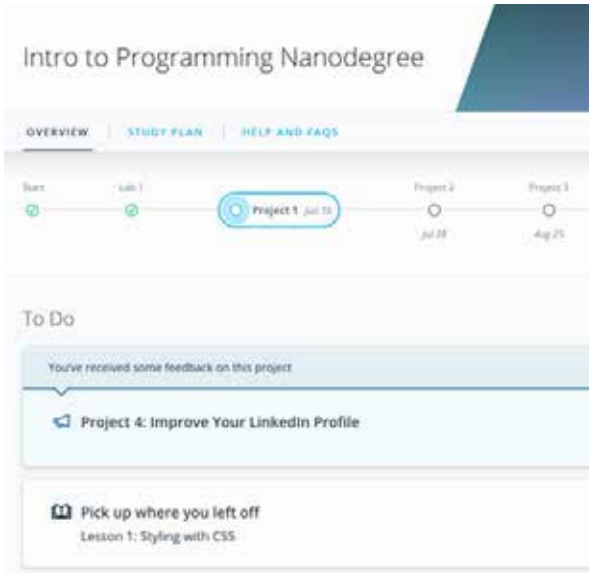
- Use Bayesian statistics to locate a robot in space.
- Learn how sensor measurements can be used to safely navigate an environment.
- Understand Gaussian uncertainty.
- Implement a histogram filter for robot localization in Python.

#### LESSON FOUR

#### Graph Slam

- Identify landmarks and build up a map of an environment.
- Learn how to simultaneously localize an autonomous vehicle and create a map of landmarks.
- Implement move and sense functions for a robotic vehicle.

# Our Classroom Experience



## REAL-WORLD PROJECTS

Build your skills through industry-relevant projects. Get personalized feedback from our network of 900+ project reviewers. Our simple interface makes it easy to submit your projects as often as you need and receive unlimited feedback on your work.

## KNOWLEDGE

Find answers to your questions with Knowledge, our proprietary wiki. Search questions asked by other students and discover in real-time how to solve the challenges that you encounter.

## STUDENT HUB

Leverage the power of community through a simple, yet powerful chat interface built within the classroom. Use Student Hub to connect with your technical mentor and fellow students in your Nanodegree program.

## WORKSPACES

See your code in action. Check the output and quality of your code by running them on workspaces that are a part of our classroom.

## QUIZZES

Check your understanding of concepts learned in the program by answering simple and auto-graded quizzes. Easily go back to the lessons to brush up on concepts anytime you get an answer wrong.

## CUSTOM STUDY PLANS

Work with a mentor to create a custom study plan to suit your personal needs. Use this plan to keep track of your progress toward your goal.

## PROGRESS TRACKER

Stay on track to complete your Nanodegree program with useful milestone reminders.

## Learn with the Best



### Sebastian Thrun

INSTRUCTOR

As the founder and president of Udacity, Sebastian's mission is to democratize education. He is also the founder of Google X, where he led projects including the Self-Driving Car, Google Glass, and more.



### Cezanne Camacho

CURRICULUM LEAD

Cezanne is an expert in computer vision with a Masters in Electrical Engineering from Stanford University. As a former researcher in genomics and biomedical imaging, she's applied computer vision and deep learning to medical diagnostic applications.



### Alexis Cook

CURRICULUM LEAD

Alexis is an applied mathematician with a Masters in Computer Science from Brown University and a Masters in Applied Mathematics from the University of Michigan. She was formerly a National Science Foundation Graduate Research Fellow.



### Juan Delgado

CONTENT DEVELOPER

Juan is a computational physicist with a Masters in Astronomy. He is finishing his PhD in Biophysics. He previously worked at NASA developing space instruments and writing software to analyze large amounts of scientific data using machine learning techniques.



## Learn with the Best



### Jay Alammar

INSTRUCTOR

Jay has a degree in computer science, loves visualizing machine learning concepts, and is the Investment Principal at STV, a \$500 million venture capital fund focused on high-technology startups.



### Ortal Arel

CURRICULUM LEAD

Ortal Arel has a PhD in Computer Engineering, and has been a professor and researcher in the field of applied cryptography. She has worked on design and analysis of intelligent algorithms for high-speed custom digital architectures.



### Luis Serrano

INSTRUCTOR

Luis was formerly a Machine Learning Engineer at Google. He holds a PhD in mathematics from the University of Michigan, and a Postdoctoral Fellowship at the University of Quebec at Montreal.

# All Our Nanodegree Programs Include:



## EXPERIENCED PROJECT REVIEWERS

### REVIEWER SERVICES

- Personalized feedback & line by line code reviews
- 1600+ Reviewers with a 4.85/5 average rating
- 3 hour average project review turnaround time
- Unlimited submissions and feedback loops
- Practical tips and industry best practices
- Additional suggested resources to improve



## TECHNICAL MENTOR SUPPORT

### MENTORSHIP SERVICES

- Questions answered quickly by our team of technical mentors
- 1000+ Mentors with a 4.7/5 average rating
- Support for all your technical questions



## PERSONAL CAREER SERVICES

### CAREER SUPPORT

- Resume support
- Github portfolio review
- LinkedIn profile optimization

# Frequently Asked Questions

## PROGRAM OVERVIEW

### WHY SHOULD I ENROLL?

The demand for engineers with computer vision and deep learning skills far exceeds the current supply. This program offers a unique opportunity to develop these in-demand skills and is for anyone seeking to launch or advance their skills in modern computer vision techniques. You'll complete several computer vision applications using a combination of Python, computer vision, and deep learning libraries that will serve as portfolio pieces that demonstrate the skills you've acquired.

### WHAT JOBS WILL THIS PROGRAM PREPARE ME FOR?

This program is designed to build on your skills in machine learning and deep learning. As such, it doesn't prepare you for a specific job, but expands your skills in the computer vision domain. These skills can be applied to various applications such as image and video processing, automated vehicles, smartphone apps, and more.

### HOW DO I KNOW IF THIS PROGRAM IS RIGHT FOR ME?

If you're new to Computer Vision, and eager to explore applications like facial recognition and object tracking, the Computer Vision Nanodegree program is an ideal choice. The curriculum introduces you to image analysis with Python and OpenCV, then goes on to cover deep learning techniques that can be applied to a variety of image classification and regression tasks. Over the course of the program, you'll leverage your Python coding experience to build a broad portfolio of applications that showcase your newly-acquired Computer Vision skills.

## ENROLLMENT AND ADMISSION

### DO I NEED TO APPLY? WHAT ARE THE ADMISSION CRITERIA?

No. This Nanodegree program accepts all applicants regardless of experience and specific background.

### WHAT ARE THE PREREQUISITES FOR ENROLLMENT?

You must have completed a course in Deep Learning equivalent to the Deep Learning Nanodegree program prior to entering the program. Additionally, you should have the following knowledge: Intermediate Python programming knowledge, including:

- Strings, numbers, and variables
- Statements, operators, and expressions
- Lists, tuples, and dictionaries
- Conditions, loops
- Generators & comprehensions
- Procedures, objects, modules, and libraries
- Troubleshooting and debugging



## FAQs Continued

- Research & documentation
- Problem solving

### Basic shell scripting:

- Run programs from a command line
- Debug error messages and feedback
- Set environment variables
- Establish remote connections

### Basic statistical knowledge, including:

- Populations, samples
- Mean, median, mode
- Standard error
- Variation, standard deviations
- Normal distribution

### Intermediate differential calculus and linear algebra, including:

- Derivatives & Integrals
- Series expansions
- Matrix operations through eigenvectors and eigenvalues

### IF I DO NOT MEET THE REQUIREMENTS TO ENROLL, WHAT SHOULD I DO?

We have a number of courses and programs we can recommend that will help prepare you for the program, depending on the areas you need to address. For example:

- [Intro to Machine Learning](#)
- [Intro to Data Science](#)
- [Intro to Programming Nanodegree program](#)
- [Data Analyst Nanodegree program](#)
- [Artificial Intelligence Programming with Python Nanodegree program](#)
- [Machine Learning Engineer Nanodegree program](#)
- [Deep Learning Nanodegree program](#)

### TUITION AND TERM OF PROGRAM

#### HOW IS THIS NANODEGREE PROGRAM STRUCTURED?

The Computer Vision Nanodegree program is comprised of content and curriculum to support three (3) projects. We estimate that students can complete the program in three (3) months working 10 hours per week.

Each project will be reviewed by the Udacity reviewer network. Feedback will be provided and if you do not pass the project, you will be asked to resubmit the project until it passes.



## FAQs Continued

### **HOW LONG IS THIS NANODEGREE PROGRAM?**

Access to this Nanodegree program runs for the length of time specified in the payment card above. If you do not graduate within that time period, you will continue learning with month to month payments. See the [Terms of Use](#) and [FAQs](#) for other policies regarding the terms of access to our Nanodegree programs.

### **I HAVE GRADUATED FROM THE COMPUTER VISION NANODEGREE PROGRAM BUT I WANT TO KEEP LEARNING. WHERE SHOULD I GO FROM HERE?**

Many of our graduates continue on to our Artificial Intelligence Nanodegree program, Natural Language Processing Nanodegree Program, Robotics Engineer Nanodegree program, and our Self-Driving Car Engineer Nanodegree programs. Feel free to explore other Nanodegree program options as well.



## SOFTWARE AND HARDWARE

### **WHAT SOFTWARE AND VERSIONS WILL I NEED IN THIS PROGRAM?**

You will need a computer running a 64-bit operating system (most modern Windows, OS X, and Linux versions will work) with at least 8GB of RAM, along with administrator account permissions sufficient to install programs including Anaconda with Python 3.5 and supporting packages. Your network should allow secure connections to remote hosts (like SSH). We will provide you with instructions to install the required software packages. Udacity does not provide any hardware.