A. Papers

This is a list of papers that tackle the DL model and attacks and defenses against self-driving cars. A summary of only a part of them can be found here, in the Autonomous Cars folder: https://github.abudhabi.nyu.edu/ha59/HK-CyberSecLab/tree/master/AutonomousCarSecurity. If you don't have access to the repo, kindly ask the professor to give you access.

Attacks and Defenses

- Vrizlynn et al. Autonomous Vehicle Security: A Taxonomy of Attacks and Defences
- Papernot et al. The Limitations of Deep Learning in Adversarial Settings
- Papernot et al. Distillation as a Defense to Adversarial Perturbations against Deep Neural Networks
- Papernot et al. Practical Black-Box Attacks against Machine Learning
- Eykholt et al. Robust Physical-World Attacks on Deep Learning Visual Classification
- Petit et al. Potential Cyberattacks on Automated Vehicles
- Petit et al. Remote Attacks on Automated Vehicles Sensors: Experiments on Camera and LiDAR
- Szegedy et al. Intriguing properties of neural networks
- Goodfellow et al. Explaining and Harnessing Adversarial Examples
- Papernot et al. Transferability in Machine Learning: from Phenomena to Black-Box Attacks using Adversarial Samples
- Grosse et al. Adversarial Perturbations Against Deep Neural Networks for Malware Classification
- Adversarial Sample code here; Nguyen et al. Deep Neural Networks are Easily Fooled: High Confidence Predictions for Unrecognizable Images
- Xu et al. Automatically Evading Classifiers: A Case Study on PDF Malware Classifiers
- Kantchelian et al. Evasion and Hardening of Tree Ensemble Classifiers
- Biggio et al. Support Vector Machines Under Adversarial Label Noise
- Biggio et al. Poisoning Attacks against Support Vector Machines
- Ororbia II et al. Unifying Adversarial Training Algorithms with Flexible Deep Data Gradient Regularization
- Jin et al. Robust Convolutional Neural Networks under Adversarial Noise
- Goodfellow et al. Deep Learning Adversarial Examples – Clarifying Misconceptions
- *Tian et al. DeepTest: Automated Testing of Deep-Neural-Network-driven Autonomous Cars - https://deeplearningtest.github.io/deepTest/
- *Rauber et al. Foolbox: A Python toolbox to benchmark the robustness of machine learning models - full documentation: https://media.readthedocs.org/pdf/foolbox/stable/foolbox.pdf
- Su, J. et al. One pixel attack for fooling deep neural networks

Other Useful Links

- https://medium.freecodecamp.org/hacking-cars-a-guide-tutorial-on-how-to-hack-a-car-5eafcfbbb7ec
- http://blog.davidsingleton.org/nnrccar/
- https://github.com/udacity/self-driving-car/tree/master/datasets
- https://www.linkedin.com/pulse/teaching-car-how-drive-using-deep-learning-muhieddine-el-kaissi/
- https://nicolovaligi.com/reading-list-udacity-self-driving-challenge-3.html
- NIPS: https://nips.cc/Conferences/2018/CompetitionTrack

Capstone Project Code

- https://www.youtube.com/redirect?q=https%3A%2F%2Fgithub.com%2FKairos-Automotive%2Fcarla-brain&redir_token=kPP2arRKn7FmCATm-OKHeMCxO6d8MTUyODE3Nzc5MkAxNTI4MDkxMzky&v=956Q7wU0-IE&event=video_description

Interesting resources to look into: **driving assistants** – MobilEye C2-270, ibo LUX 3

B. Simulators

1. Udacity

This is the first simulator I would recommend you to start with, as it is fairly simple, as you will see.

No operating system preferred.

Installation

- Github repository**:** https://github.com/udacity/self-driving-car-sim
- Link to a repository that explains the installation steps: https://github.com/llSourcell/How_to_simulate_a_self_driving_car
- Installation video: https://www.youtube.com/watch?v=EaY5QiZwSP4&index=3&list=PLSRMSHOzuuvAJJEx49s7-IzR7L0fOGHSv&t=0s
- More resources on how the deep learning model works: https://github.com/naokishibuya/car-behavioral-cloning
- Paper for the NVIDIA model (that the simulator is based on): https://arxiv.org/pdf/1604.07316v1.pdf

2. Apollo Auto

You need Linux to run this simulator.

Although it is more complex than Udacity, I didn't find it particularly useful. However, the GPS architecture is worth looking into.

Official website: http://apollo.auto

Installation

- Github repository: **https://github.com/ApolloAuto/apollo**

3. Carla

I recommend Linux to run this simulator.
By far, the most complex simulator, it looks very much like a video game that has both training and autonomous modules.

Installation

- Github repository: https://github.com/carla-simulator/carla
- How to build on Linux: http://carla.readthedocs.io/en/latest/how_to_build_on_linux/
- How to build on Windows: http://carla.readthedocs.io/en/latest/how_to_build_on_windows/
- Paper: http://carla.org http://proceedings.mlr.press/v78/dosovitskiy17a/dosovitskiy17a.pdf
- To run the autonomous mode: first run PythonClient and then run the server

C. Attack Models

- Camera spoofing
- GPS spoofing

D. Courses

- Coursera, Machine Learning: https://www.coursera.org/learn/machine-learning (**it has just started on the 11th, it's a good opportunity for you to enroll now)**
- Google, Deep Learning https://ae.udacity.com/course/deep-learning--ud730
- MIT, Autonomous Cars course: http://selfdrivingcars.mit.edu (very good class that I recommend)
  Berkeley: https://deepdrive.berkeley.edu/node/107

E. Presentations & Reports

If you don't have access to the repo, ask the professor to grant you access.

- https://github.abudhabi.nyu.edu/ha59/HK-CyberSecLab/tree/master/AutonomousCarSecurity