ECSE 4965/6965

Introduction to Deep Learning

Spring, 2018

Instructor: Dr. Qiang Ji,
Email: jiq@rpi.edu
Phone: 276-6440
Office: JEC 7004
Meeting Hours & Place : 4:00-5:20 pm, Mondays and Thursdays, EATON 214
Office Hours: 5:30-6:30 pm Mondays and Thursdays or by Appointment
TAs:

Keyi Liu liuk7@rpi.edu
Javad Heydari heydas@rpi.edu

Lecture notes: Available on RPI Learning Management System

This course introduces fundamentals in deep learning and demonstrates its applications in computer vision. It covers both deterministic and probabilistic deep models. Deterministic deep models include Deep Neural Networks, Convolutional Neural Networks, Recurrent Neural Networks, Auto-encoders, and the Generative Adversarial Networks. Probabilistic deep models include Bayesian Neural Networks, Deep Boltzmann Machine, Deep Belief Networks, and Deep Bayesian Networks. In addition, the course will also cover the latest deep developments in deep reinforcement learning. The course is self-contained. It starts with an introduction of the background needed for learning deep models, including probability, linear algebra, standard classification and optimization techniques. To demonstrate various deep models, we will apply them to different computer vision tasks, including object recognition, human action recognition, and facial expression recognition.

Prerequisites

This is a senior and graduate level course. Students should have basic knowledge in probability, linear algebra, and optimization. Strong programming skills in one of the high level languages such as C++, Matlab, or Python are required. Prior courses in machine learning/ pattern recognition and computer vision/image processing are preferred but not required.

Optional textbook

Deep Learning, Ian Goodfellow and Yoshua Bengio and Aaron Courville, MIT Press, 2016, http://www.deeplearningbook.org (download the book quickly for free!)

Software

Many of the assignments and projects will be implemented in Google's TensorFlow (https://www.tensorflow.org/tutorials/), which is implemented in Python. Mastery of Python or being able to quickly learn it is required.

Course Evaluation

The course will involve homework assignments (10%), class projects (50%), a mid-term exam (20%), and a final project (20%). All exams are open book and comprehensive.