CS4910: Deep Learning for Robotics

David Klee
klee.d@northeastern.edu

T/F, 3:25-5:05pm
Behrakis Room 204

https://www.ccs.neu.edu/home/dmklee/cs4910_s22/index.html

https://piazza.com/northeastern/spring2022/cs4910a/home

Guidance for Successful Project

Dedicate a fraction of your time to research/reading

- Find alternative methods that address your problem
- Read papers about the method you are working on
- If you come across an idea and are unsure if it applies, I can help

Create minimum viable simulator for training agent or collecting data

- Implement basic functionality (including **rendering!**)
- Avoid realism, it should be fast and minimal; only use Pybullet if you need to
- No need to be generalizable: if you are unsure about action space, select one arbitrarily

Do not waste time on Pybullet, ask for help. You should be able to finish this within a few hours. Collect a video of it working

Implement learning agent

- Implement single agent that will work with your MVS
- Create infrastructure that will be used for the rest of the project
  - Repeat trials, plotting, logging, storing videos/models
- What compute resources are best (your laptop, Colab, etc.)
- Identify any bugs in MVS
- Once it is working, search over major decision variables (i.e. state space, action space, loss function), this may require building on MVS

Hypothesis-driven vs. grid-search

CNN vs MLP

Number of layers in CNN

Ideal batch size

Reward Scheme

Size of action space

Hypothesis-driven vs. grid-search

CNN vs MLP

Reward Scheme

Number of layers in CNN

Ideal batch size

You should leverage computing as much as possible, but do not blindly run experiments. Hypothesis-driven development will lead to more insights and help you hone your understanding of deep learning

Working with real-robot will take longer than you think

My belief: a sub-optimal agent that works on the real-robot is more impressive than an optimal agent in the simulator

Please keep me in the loop about real-world dependencies. It may take a few days to order/3D-print parts or build setups

Expectations for In-class Presentations

- Upload slides or a report before class (there will be piazza post)
  - 3-5 minutes of content
  - This should **not** be a data-dump!
  - Include details about action/state-space, etc.
  - Quantitative vs Qualitative results
  - Bad results can be informative

- Engage in discussion about classmates work

Keep your sights set on the final deliverable

What figures/videos will make for a compelling website or report

If publishing code, develop habit of readable, well-documented code

Next Class (3.8.22)

Mastering the game of Go with deep neural networks and tree search

Is everyone around next friday?