BME 548L: Machine Learning and Imaging - Spring 2022 Syllabus

Class times: Monday and Wednesdays 3:30pm – 4:45pm
Physical class location for lectures: Hudson 125
Online class lecture location: https://duke.zoom.us/j/91219140163

Lab times: Mondays *or* Wednesdays at 5:15pm-6:30pm
Physical class location for labs: Hudson 125
Online class location for labs: https://duke.zoom.us/j/92166840230

Instructor:
Roarke Horstmeyer - rwh4@duke.edu
Office hours: Tuesdays 11:00am - 12:00pm
Office hours: Wednesdays 11:00am - 12:00pm
Office hours location: Zoom Meeting Link: https://duke.zoom.us/j/91219140163
(Same link as used for the lecture)

Teaching Assistants:
Kanghyun Kim – kanghyun.kim@duke.edu
Shiqi Xu – shiqi.xu@duke.edu

Course website: **deepimaging.github.io**
Slack: **deepimaging.slack.com**

Introduction:
Welcome to Machine Learning and Imaging, BME 548L! This class is an overview of machine learning and imaging science, with a focus on the intersection of the two fields. This class is for you if 1) you work with imaging systems (cameras, microscopes, MRI/CT, ultrasound, etc.) and you would like to learn more about machine learning, 2) if you are familiar with machine learning and would like to know more about how your data is gathered, 3) if you work with both imaging systems and machine learning and would like to hear a new perspective on the topic, or 4) if you work with neither imaging systems nor machine learning but have a really strong mathematical and signal processing background and are motivated to learn about both.

Goals:
By the end of this course, my aim is for you to be comfortable with the following:

1) Understand the core mathematical concepts underlying machine learning
2) Understand the detailed operation of convolutional neural networks
3) Understand how to model and simulate various imaging systems
4) Understand how to merge imaging system models into machine learning frameworks
5) Be able to write your own machine learning code for image data analysis and/or system design

Course structure:
This course is primarily designed for Masters and PhD students who wanted to learn more details about a current topic of active research. I've also opened the class up to advanced undergraduates (primarily Seniors, but anyone with sufficient programming and mathematics background should be fine). I have attempted to take into account a spread of experience levels while re-designing the material and I'll try to keep that in mind as I'm lecturing.

This class assumes a certain level of background knowledge in math and programming (see pre-requisites below). It will be relatively fast-paced and will skip over some details to reach its primary goal, which is to help each student identify and work on a suitable final project. The final project should be something that you are excited about and could certainly be related to your current research. If you are not currently pursuing a related research topic or any research topic, then that is ok – we can work together to find a suitable final project topic. A very good outcome of this course will be if each student can write machine learning code that they fully understand, that tests something of interest to them (i.e., not just classifying images of cats and dogs), and that includes some hypothesis-driven component to it.

Lab sections: This class has a lab component, held on Mondays and Wednesdays 5:15pm-6:30pm on Zoom. You only need to attend one lab session per week (unless you really want to attend both). Labs will be focused on the coding aspects of this class. The TAs will do his best to teach you how to write machine learning code in Python/Tensorflow, review topics similar to problems with the homework, and provide assistance with final projects towards the end of the class.

Pre-requisites:
- Linear algebra – vectors, matrices, tensors, dimensional analysis (MATH 221 or equivalent)
- Signal processing– Linear systems, convolutions, Fourier transforms (BME 271 or equivalent)

- Imaging and instrumentation (BME 303 or equivalent)
- Programming – MATLAB, basic Python (Numpy, Scipy), Tensorflow 2.0

Communication:

1) **deepimaging.github.io – This is the main course website**
2) Slack: **deepimaging.slack.com** - I hope you will feel comfortable asking questions, posting comments and sharing insights here. The TA's and I will actively communicate with you all here.
3) Google Co-Lab: We will use Google Co-Lab for coding assignments. More information on that will be provided in the first lab sessions.
4) Jupyter notebooks: We will also use and encourage the use of Jupyter notebooks to test and share code.
5) Sakai: I use Sakai to post grades (sakai.duke.edu). I'll likely post the homework assignments up there as a back-up, but that's about it.

Programming assignments: This course will use Python for programming assignments. Some background knowledge of Python will be required (or, an in-depth knowledge of MATLAB will likely be sufficient, since many MATLAB "skills" translate nicely).

Homework assignments: There will be 5 homework assignments throughout the semester. These assignments will be part problem-based and part code-based. Collaboration on assignments is encouraged, but I expect each student to write their own solutions in their own way, and to not directly copy code or code segments.

Homework policy: Homework will be due by 11:59pm on the stated date and can be submitted via Github and/or email (still TBD the best way to manage this). Late assignments will receive a 20% lower score for each late day (no fractional days).

Quizzes: There may be a quiz or two during the semester. Quiz dates will be announced beforehand. Each quiz will be 30-45 minutes. Collaboration on quizzes is not allowed.

Missed quiz policy: Missed quizzes will receive a 0 unless a Deans excuse is provided. Students who submit a Deans excuse can make-up a similar quiz with different content.

Final project: A large component of the course will be for each student to code-up a machine learning framework that can help answer a relevant scientific question. Students will submit topic proposals that they will receive feedback on before final project work begins. Students may complete the final project individually or in small groups (the expected amount of effort/accomplishment will scale with the size of the group). The final project will consist of submitting the following sub-components:

1) The source code and data that you used (if you're allowed to share it)
2) A short research-style paper (4 pages minimum, 6 pages maximum) that includes an introduction, results, a discussion and some figures and references
3) An 8-minute presentation that each student will deliver via a Zoom presentation during one of several presentation slots that we'll setup during the finals period.

Participation and engagement: Participation is encouraged in this class. The semester participation grade will be self-evaluated on a scale of 0-5, both at the middle and the end of the semester. Each time, you should prepare a brief 1-paragraph explanation of why you deserve the score you selected (e.g., times you asked questions, provided answers, posted things on Piazza). I can choose to accept or reject your selected score. Note that merely attending class does not earn you a 5.

Lecture: I expect you to show up to lecture as much as possible. I encourage questions during lecture, and you should feel free to ask any question, no matter how simple it may seem. This is important – do not feel like you cannot ask simple questions, because these are usually the most important ones. However, this material is quite complex, so I am going to reserve the right to put off some questions until later/after lecture to make sure we stay on schedule.

Office hours: You should feel free to join me on Zoom (see Zoom Meeting ID at beginning of syllabus) to ask questions. TA's will have their own separate office hours (TBA).

Collaboration: You must adhere to the Duke Community Standard in all work you do for this course. Please read this and be familiar with it. I am going to encourage you guys to work together on homeworks and programming assignments. While the earlier assignments will be more geared towards ensuring everyone has some foundational knowledge, later assignments will cover relatively recent topics in machine learning and imaging. These later assignments will be exploratory and will benefit from collaboration. You may **not** collaborate on the quizzes. Collaborating on these will be a violation of the community standard.

Grading:
Your final grade will be determined via the following breakdown:
Homework: 45%
Final project: 40%

Project proposal: 7%
Participation: 8%

Resources:
This class will not closely follow a book, since (to the best of my knowledge) there aren't any books that cleanly teach these topics yet. Here are a few that should be helpful throughout this class:
Deep Learning, A. Goodfellow et al.: https://www.deeplearningbook.org/
Introduction to Fourier Optics, J. Goodman
Learning from Data, Y. S. Abu-Mustafa
Introduction to Linear Algebra, G. Strang

And here are a few other classes that have some very helpful slides and lectures:
Stanford CS231n: http://cs231n.stanford.edu/syllabus
Caltech, Learning from Data: https://work.caltech.edu/telecourse.html
Stanford CS230: http://cs230.stanford.edu/syllabus

Tentative Course schedule:
Week 0 – Jan 5: Machine learning and imaging systems in a nutshell
Week 1 – Jan 10, 12: Review background mathematics – linear algebra, etc.
Week 2– Jan 17 (Holiday, no class), Jan 19: Optimization and cost functions
Week 3– Jan 24, 26: From optimization to machine learning
Week 4 – Jan 31, Feb 2: Neural networks, the chain rule and back-propagation
Week 5 – Feb 7, 9: Convolutional neural networks (CNN's)
Week 6 – Feb 14, 16: CNN's in practice
Week 7 – Feb 21, 23: Extended applications of CNN's
Week 7 – Feb 28, March 2: Light propagation and imaging systems
Week 8 – March 7, 9: No Class
Week 9 – March 21, 23: Computational models of imaging systems
Week 10 – March 28, 30: Project proposals and discussions
Week 11 – April 4, 6: Designing imaging systems with CNN's
Week 12 – April 11, 13: Reinforcement Learning, Gen. adversarial networks
Final project presentations: Wednesday April 27, 7-10pm (class finals slot), details TBD.