BME 548L: Machine Learning and Imaging - Spring 2024 Syllabus

Class times: Monday and Wednesdays 3:05pm – 4:20pm Physical class location for lectures: Hudson Hall 207 Online class lecture location: <u>https://duke.zoom.us/j/93342721843</u>

Lab times: Mondays *or* Wednesdays at 4:40pm-5:55pm Physical class location for labs: Hudson Hall 207 Online class location for labs: TBD

Instructor:

Roarke Horstmeyer - <u>rwh4@duke.edu</u> Office hours: Tuesdays 11:00am - 12:00pm Office hours location: Zoom Meeting Link: <u>https://duke.zoom.us/j/93342721843</u> (Same link as used for the lecture)

Teaching Assistants: Xi Yang- <u>xy95@duke.edu</u> Amey Chaware - <u>amey.chaware@duke.edu</u>

Adjunct Instructor: Lucas Kreiss - <u>lucas.kreiss@duke.edu</u>

Important – we'll be using the following course website to distribute material: Course website: deepimaging.github.io

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. The class is also suitable for 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 4:40pm-5:55pm. 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) Canvas- We'll try to make announcements and accept homework assignments and grade them via Canvas. We are all new to Canvas, so please bear with us as we figure out how to best use it. If it doesn't work well, we might switch to Slack (TBD).
- 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.

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.

Lab workbook assignments: 1 week before many labs, a lab workbook will be shared via the course webpage (under the TA Sessions tab) and via Sakai. Students must attempt and complete the 1-4 questions within these workbooks and submit these workbooks via Sakai to receive a Pass/Fail credit on each workbook.

Homework policy: Homework will be due by 11:59pm on the stated date and can be submitted via Sakai. 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 and 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 office hours (TBA).

Collaboration: You must adhere to the <u>Duke Community Standard</u> in all work you do for this course. Please read this and be familiar with it. I am going to encourage everyone to work together on homeworks and programming assignments, but **please do not directly copy each other** – instead, learn from each other, but please formulate your own code and own solutions. If you work closely together with one or more students, please cite who you worked with at the top of your submission. 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.

Note about large language models (LLMs): As a course about machine learning, we are of course excited about how new software methods can be used to assist with learning and software development. That said, this is a course where the main objective is to *learn*. Accordingly, it is fine to use LLMs like ChatGPT and Bard help with your learning, but just like above, please **please do not directly copy from LLMs**. Please formulate your own code and own solutions. **If you work closely together with and LLM, please cite which one you worked with at the top of your submission**.

Grading:

Your final grade will be determined via the following breakdown: Homework: 42% Final project: 38% Project proposal: 7% Lab Workbooks: 5% 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.: <u>https://www.deeplearningbook.org/</u> 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: <u>http://cs231n.stanford.edu/syllabus</u> Caltech, Learning from Data: <u>https://work.caltech.edu/telecourse.html</u> Stanford CS230: <u>http://cs230.stanford.edu/syllabus</u>

Tentative Course schedule:

Week 0 – Jan 10: Machine learning and imaging systems in a nutshell Week 1 – Jan 17 (Holiday, no class), Jan 18: Review background mathematics Week 2– Jan 22, Jan 24: Linear algebra, optimization and cost functions Week 3– Jan 29, Jan 31: From optimization to machine learning Week 4 – Feb 5, 7: Neural networks, the chain rule and back-propagation Week 5 – Feb 12, 14: Convolutional neural networks (CNN's) Week 6 – Feb 19, 21: Extended applications of CNN's Week 7 – Feb 26, 28: Light propagation and imaging systems Week 8 – March 4, 6: Computational models of imaging systems Week 9 – March 11, 13: Spring Break, No Class Week 10 – March 18, 20: Project proposals and discussions Week 11 – March 25, 27: Designing imaging systems with CNN's Week 12 – April 1, 3: System Design and Optimization with CNN's Week 13 – April 8, 10: Object Detection, Autoencoders Week 14 - April 15, 17: Reinforcement Learning, Gen. adversarial networks Final project presentations: Thursday May 2, 7pm-10pm (class finals slot), details TBD. Tentative Schedule of Assignments: HW1 assigned: Jan 29 HW1 due: Feb 12 HW 2 assigned: Feb 12 HW2 due: Feb 28 HW3 assigned: March 4

Final project proposal due: March 8

HW3 due: March 13

HW4 assigned: March 18

HW4 due: April 1

HW5 assigned: April 1

HW 5 due: April 15