

Course Syllabus

Methods in Biomedical Image Analysis

Spring 2020

Dr. John Galeotti: jgaleotti@cmu.edu, CMU Robotics & Pitt Bioengineering

Office Hours: after class as needed, or by appointment

Course Description: Students will gain theoretical and practical skills in 2D, 3D, and 4D biomedical image analysis, including skills relevant to general image analysis. The fundamentals of computational medical image analysis will be explored, leading to current research in applying geometry and statistics to segmentation, registration, visualization, and image understanding. Additional and related covered topics include de-noising/restoration, morphology, level sets, and shape/feature analysis. Students will develop practical experience through projects using the latest version of the National Library of Medicine Insight Toolkit (ITK) and SimpleITK, a popular open-source software library developed by a consortium of institutions including Carnegie Mellon University and the University of Pittsburgh. In addition to image analysis, the course will include interaction with radiologists and pathologist(s). *** Lectures are at CMU and students will visit clinicians at UPMC. Some or all of the class lectures may also be videoed for public distribution, but students may request to be excluded from distributed video. Website: http://www.cs.cmu.edu/~galeotti/methods_course/

Prerequisites: Knowledge of vector calculus, basic probability, and either C++ or python, including basic command-line familiarity and how to pass arguments to your own command-line programs. Extensive expertise with C++ and templates is not necessary, but some students may find it helpful.

Objectives: To gain theoretical and practical skills in biomedical image analysis algorithms.

Location: CMU (primary) & UPMC (secondary)

Required Textbook: "Machine Vision", ISBN: 052116981X

Optional Textbook: "Insight to Images", ISBN: 9781568812175

Course Website: http://www.cs.cmu.edu/~galeotti/methods_course/

You should check the course website daily for announcements and handouts.

Tentative Course Calendar: A tentative course calendar is listed on the course website (**see above**). The lecture schedule (and some topics) are subject to change, depending in part on class interest and involvement.

Grading Algorithm: The point system

- Each question or problem in a quiz or homework is assigned a point value

- Your cumulative grade for quizzes [or homeworks] is (the sum of points your earned on all quizzes) divided by (the sum of points you could have earned on all quizzes)
- So, your course grade is equally affected whether you miss 1 point on a 3 point quiz, or you miss 1 point on a 10 point quiz. (This is not the case for the more typical "averaged percentages" method.)

Attendance: Required

- Checked using Quizzes
- On some days the quiz may be signing your name on the roll.
- Some days may not have any quiz at all (attendance not checked).

Quizzes: 20%

- Not present / not taken = 0
- Lowest 2 are dropped (the 2 on which you missed the most points)
- So, if you are gone for a week-long conference, then the 2 0's won't count.
- In case of extenuating circumstances requiring further absence, talk to me, but I must be fair to the class (i.e. harder on you).

Homework: 30%

- Your TA will help you *before* the assignment is due. When grading, he will **not** try to figure out a non-working mess of code!
- **Late policy: 0%** for code that does not compile, run, and at least perform some part of the assignment. *However*, if you've made a reasonable effort in advance *and* have been working with the TA but still have not been able to get things to work, then we will be much more generous with partial credit and/or extra time, on a case-by-case basis.
- Also, if you using a different compiler than the TA, then you will be given a brief period of time to fix unforeseen cross-platform incompatibilities.

Clinical Shadow Program: 10%

- You submit 1 report for each clinical station you visit.
- The first time you miss a station for which you are scheduled (without good reason), you may contact your instructor to reschedule for 50% credit for that station.
- If you do not show up a second time, you will be removed from the Shadow Program, get a 0 for *all subsequent* stations, and your instructor will be very unhappy with you.

Final Project: 40%

- ◦ 15% presentation (see below for "Communication & Presentation Help")
- ◦ 25% code

Final Letter Grade

- While lower cutoffs may be used, the following maximum grade cutoffs are guaranteed:

>= 93.5 A
 >= 90.0 A-
 >= 87.5 B+
 >= 83.5 B
 >= 80.0 B

Accommodations for Students with Disabilities:

If you have a disability and have an accommodations letter from the Disability Resources office, I encourage you to discuss your accommodations and needs with me as early in the semester as possible. I will work with you to ensure that accommodations are provided as appropriate. If you suspect that you may have a disability and would benefit from accommodations but are not yet registered with the Office of Disability Resources, I encourage you to contact them at access@andrew.cmu.edu.

Mental Health

Just as you optimized your code and assignments, you will most likely at some point benefit from your own mental optimization. As a student, you may experience a range of challenges that can interfere with learning, such as strained relationships, increased anxiety, substance use, feeling down, difficulty concentrating and/or lack of motivation. These mental health concerns or stressful events may diminish your academic performance and/or reduce your ability to participate in daily activities. CMU services are available, and they work. You can learn more about confidential mental health services available on campus at: <http://www.cmu.edu/counseling/>. Support is always available (24/7) from Counseling and Psychological Services: 412-268-2922.

Academic Integrity and Collaboration

Honesty and transparency are important features of good scholarship. On the flip side, plagiarism and cheating are serious academic offenses with serious consequences. If you are discovered engaging in either behavior in this course, you will earn a failing grade on the assignment in question, and further disciplinary action may be taken.

For a clear description of what counts as plagiarism, cheating, and/or the use of unauthorized sources, please see the University's Policy on Academic Integrity (revised in April 2013): [http://www.cmu.edu/policies/documents/Academic Integrity.htm](http://www.cmu.edu/policies/documents/Academic%20Integrity.htm)

I encourage you to work together on homework assignments and to make use of campus resources like Academic Development, the Global Communication Center, and the Intercultural Communication Center to assist you in your pursuit of academic excellence. However, please note that in accord with the university's policy you must acknowledge any collaboration or assistance that you receive on work that is to be graded: so when you turn in a homework assignments, please include a sentence at the end that says either:

1. "I worked alone on this assignment.", or
2. "I worked with _____ on this assignment." and/or
3. "I received assistance from _____ on this assignment."

Note that providing this information will only serve to help me understand you better: I strongly endorse the use of campus resources like Academic Development and the Global Communication Center, as well as collaborative learning, when it increases your ability to succeed in this class and when it enhances your education and learning.

If you have questions about my integration of the university's policy into this course, please do not hesitate to ask: my aim is to foster an environment where you can learn and grow, while ensuring that the work we all do is honest and fair. For more information about Carnegie Mellon's standards with respect to academic integrity, you can also check out the following link: <http://www.cmu.edu/academic-integrity/>

Here are some examples of acceptable collaboration:

- Clarifying ambiguities or vague points in class handouts, textbooks, or lectures.
- Discussing or explaining the general class material.
- Providing assistance with Python/C++, in using the system facilities, or with editing, debugging, and programming tools and libraries.
- Discussing the code that we give out on the assignment.
- Discussing the assignments to better understand them.
- Getting help from anyone concerning programming issues which are clearly more general than the specific assignment (e.g., what does a particular error message mean?).

Now for the dark side. As a general rule, if you do not understand what you are handing in, you are probably cheating. If you have given somebody the answer, you are probably cheating. In order to help you draw the line, here are some examples of clear cases of cheating:

- Copying (program or assignment) files from another person or source, including retyping their files, changing variable names, copying code without explicit citation from previously published works (except the textbook), etc.
- Allowing someone else to copy your code or written assignment, either in draft or final form.
- Getting help from someone whom you do not acknowledge on your solution.
- Copying from another student during an exam, quiz, or midterm. This includes receiving exam-related information from a student who has already taken the exam.
- Writing, using, or submitting a program that attempts to alter or erase grading information or otherwise compromise security.
- Inappropriately obtaining course information from instructors and TAs.
- Looking at someone else's files containing draft solutions, even if the file permissions are incorrectly set to allow it.
- Receiving help from students who have taken the course in previous years.
- Lying to course staff.
- Copying on quizzes or exams.
- Reviewing any code submissions from previous years.
- Reading the current solution (handed out) if you will be handing in the current assignment late.

Education Objectives (Relationship of Course to Program Outcomes)

(a) an ability to apply knowledge of mathematics, science, and engineering: Theoretical and practical lectures are combined with practical and empirical exercises to prepare students for a large-scale science/engineering final project.

(b) an ability to design and conduct experiments, as well as to analyze and interpret data: Class exercises require students to design, build, and run experiments in software to empirically optimize their projects' algorithm architecture and parameter tuning.

(c) an ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability: The final projects require students to build a working system that address a real need, either by providing new computational tools or by making new discoveries. Required interaction with practicing clinicians and class lectures both expose students to the dominant constraints in the various domains of biomedical image analysis, including matters of usability, patient safety, and legal liability.

(d) an ability to function on multi-disciplinary teams: The entire class is multi-disciplinary, requiring students to interact with clinicians, biologists, and/or engineers. Homework assignments and the final project require engineering approaches to biomedical problems in order to derive clinically/scientifically meaningful results.

(e) an ability to identify, formulate, and solve engineering problems: As a project-based course, students must continually solve engineering problems. The final projects further require students to individually identify relevant biomedical problems, formulate an engineering approach, and then proceed to (at least partially) solve their chosen problems

(f) an understanding of professional and ethical responsibility: Ethical matters of patient safety and legal liability are discussed when relevant to other lecture materials, as part of projects, and when shadowing clinicians.

(g) an ability to communicate effectively: The final projects require individual student presentations, roughly equivalent to an oral conference presentation. Although students are graded primarily on technical content, they are also graded on the clarity, polish, and length of their presentation.

(h) the broad education necessary to understand the impact of engineering solutions in a global, economic, environmental, and societal context: [see item (c) above regarding economic and societal constraints and impact]

(i) a recognition of the need for, and an ability to engage in life-long learning The course repeatedly stresses the need to always consult the latest scientific literature when seeking to build useful systems, and makes use of current papers for some of the lectures. The textbook was chosen in part due to its extensive references which provide an excellent cross-reference starting point from which relevant current literature can be found.

(j) a knowledge of contemporary issues: Biomedical image analysis is a driving factor for many recent developments in the medical and biological communities. Students are instructed and shown that building useful systems requires knowledge of contemporary practice, workflow,

and limitations within these communities, so that the students' final systems will be both usable and relevant to current research/medicine.

(k) an ability to use the techniques, skills, and modern engineering tools necessary for engineering practice: As a project based course, students are taught and required to use a wide variety of techniques, skills and engineering tools to solve a variety of biomedical image analysis problems.