Computer Vision

CS 682 | Fall 2020 | Professor Gregory J. Stein

Course Information

CS 682
3 Credits
Lectures: Tue 7:20 PM –10:00 PM

Instructor Information

Prof. Gregory J. Stein
He, Him, His
gjstein@gmu.edu

Course Description

The aim of computer vision is to compute properties of the three-dimensional world from images so that it may be understood by machines. Topics in this class include how to create panoramic images, to build a 3D reconstruction of an environment from videos, and to recognize familiar people and objects, all through analysis of images and video clips.

This course is at the introductory graduate level and aims to introduce students to the field of computer vision through hands-on programming projects. By the end of the course, students should be familiar with the standard tools used by the computer vision community, and be able to read and understand research papers in the field.

See the detailed class schedule below for more details on what will be covered during the course.

Course Structure

Especially because this course will be taught online (and as a single 2h40m lecture on Tuesday evenings), ensuring that the students are engaging with the material and with one another can be difficult. As such, most lectures will be broken into subparts: each class will consist of two ~50 minute lectures, each followed by a 20 minute breakout session during which students will collaborate to solve short programming assignments related to the course material. Remaining class time will consist of a shorter lecture covering a related topic or research area.

Prerequisites

- Algorithms and Data Structures
- Artificial Intelligence
- Linear Algebra and Calculus

In addition, students are expected to have some familiarity with the Python programming language (including numpy) in which all programming projects and assignments will be done.

Assignments & Grading

The grading in this course will come from three main components:

- 7 programming assignments (70%)
- 8 take home "quizzes" (24%)
- Participation (6%)

All assignments will be turned in via Blackboard.

Programming Assignments

In each programming assignment, students will be expected to implement some of the algorithms we will discuss during class. Assignments will be given in the form of Jupyter Notebooks and will often include some partially written code for students to complete.

Students will be expected to write up a report for each programming assignment in LaTeX. As this is an introductory graduate course, some creativity on the solutions is expected; some of the problems will have open-ended prompts and students should expect to explore the parameter space of the algorithms they implement and report on their findings.

Students may collaborate on the programming assignments in small groups (no more than 3 or 4 students) but solutions must be written up independently. Students have a total of **6 free late days (total)** to use on programming assignments, during which there will be no penalty; any additional late days will result in a 10%/day penalty.

Quizzes

To supplement the programming assignments, which test mostly practical knowledge, the quizzes will be shorter and ask two or three conceptual questions related to the course material. Solutions must be typed using LaTeX. Students may collaborate on the quizzes in small groups (no more than 3 or 4 students) but solutions must be written up independently.

There will be a total of 8 take home quizzes assigned during the semester. Quizzes **cannot be turned in late** without prior discussion with the instructor. At the end of the term the lowest two quiz scores (including missed quizzes) will be dropped.

Participation & Lectures

Lectures will regularly include *breakout sessions* during which students will be expected to work collaboratively to tackle some small problem related to the lecture. As such, I have chosen to include participation as a small part of the overall grade to encourage students to come to class. I will not be regularly taking attendance—so missing a lecture or two without prior notice is acceptable—though students who regularly miss lecture will be given lower participation grades.

Course Resources

Textbook & Readings

Textbook readings will be assigned from Computer Vision: Algorithms and Applications, by Richard Szeliski (available as a PDF online for free). Additional supplemental readings, usually in the form of papers will be included either as links in the lecture slides or as PDFs uploaded to the course Piazza.

Lecture Slides

I will be providing lecture slides as PDF documents after each lecture via Piazza.

Discussion

This term we will be using Piazza for class discussion. The system is highly catered to getting you help fast and efficiently from classmates and myself. Rather than emailing questions to the teaching staff, I encourage you to post your questions on Piazza. Find our class signup link at: https://piazza.com/gmu/fall2020/cs682

Syllabus and Detailed Course Schedule

The following syllabus is *tentative* and subject to change.

The course is largely broken into three Units:

1. **Images** (Lectures 1–5) In which we will discuss the fundamental mathematical tools used to process images and identify common *features* between multiple images
2. **Structure** (Lectures 6–10) In which we will study how we might understand and reconstruct the 3D world.
3. **Modern Applications** (Lectures 11–14) Devoted to discussing state-of-the art research tools and applications, including Place Detection, Simultaneous Localization and Mapping, and Convolutional Neural Networks (and applications).

LEC	DATE	TOPICS	PROJECTS	QUIZES
1	8/25	Introduction Fundamentals & Image Filtering	P1 Out	
2	9/1	Fourier Transforms Resampling & Image Pyramids		Q01 Out
3	9/8	Feature & Corner Detection	P1 Due	Q01 Due

		Feature Invariance The Hough Transform (optional)	P2 Out	Q02 Out
4	9/15	Image Transformations Feature Descriptors		Q02 Due
5	9/22	Image Alignment RANSAC	P2 Due P3 Out	Q03 Out
6	9/29	Light Camera Models		Q03 Due Q04 Out
7	10/6	Panoramas Single-view Modeling	P3 Due P4 Out	Q04 Due Q05 Out
–	10/13	(No Class; Fall Break)		
8	10/20	Two-view Geometry Stereo Vision	P5 Out	Q05 Due
9	10/27	Optical Flow Tracking	P4 Due	
10	11/3	Structure from Motion Multi-View Stereo Applications Simultaneous Localization and Mapping	P5 Due P6 Out	Q06 Out
11	11/10	Image Classification Bag of Words Methods		Q06 Due Q07 Out
12	11/17	Neural Networks	P6 Due P7 Out	Q07 Due Q08 Out
13	11/24	[to be determined; likely voted on in class]		
14	12/1	Image Segmentation Course Summary		Q08 Due

Inclusion & Integrity

I stand by Mason's commitment to diversity and inclusion and hope to foster an inclusive environment in which all feel welcome in my class.

> *True diversity is defined not only as differences in individual backgrounds, personal identities, intellectual approaches, and demographics; it is also the removal of barriers and the creation of space that allow individuals to fully engage in the life of the university.*

Honor Code Statement

The GMU Honor Code is in effect at all times. In addition, the CS Department has further honor code policies regarding programming projects, which are detailed here. Any deviation from the GMU or the CS department Honor Code is considered an Honor Code violation.

Disability Accommodation

If you have a documented learning disability or other condition which may affect academic performance, make sure this documentation is on file with the Office of Disability Services and then discuss with the professor about accommodations.

Mental Wellness

Graduate School can be a stressful environment and the realities of remote work can amplify these stresses. My “door” is always open; if you are struggling with the course work or would like someone to talk to, feel free to reach out to me. GMU also provides many mental health resources that I encourage you to look at.