**CSci 4270 and 6270**
**Computational Vision**
**Spring Semester, 2021**
**Course Syllabus**

| | |
|---|---|
| Professor: | Chuck Stewart |
| Class: | Monday, Thursday, 2:30 - 4:20 |
| Office Hours: | Mon. 4:30-5:30, Wed. 1:30-3:30 |
| Professor WebEx: | `https://rensselaer.webex.com/meet/stewart` |
| Email: | `stewart@rpi.edu` |
| | |
| TA: | Alex Mankowski; `mankoa@rpi.edu` |
| TA Hours: | Tues 3:30-5:30pm, Fri 12:30-2:30pm |
| TA WebEx: | `https://rensselaer.webex.com/meet/mankoa` |
| | |
| Mentor: | Yue Han; `hany4@rpi.edu` |
| Mentor Hours: | Thurs 5:00-7:00pm |
| Mentor WebEx: | `https://rensselaer.webex.com/meet/hany4` |
| | |
| Submitty URL: | `https://submitty.cs.rpi.edu/s21/csci4270/` |

Note: all course material will be posted on the Submitty site.

## Overview

Cameras and digital images are everywhere, with billions of cameras being used to take trillions of images, producing far more data than humans can possibly absorb. The result is a wide-spread need for algorithms and software that can combine, summarize and interpret these images, a need for the fields of computer vision and image analysis. Applications of computer vision include face, object and scene recognition, security and biometrics, photomontaging and virtual tours, special effects in graphics, photography and the movies, autonomous robots, self-driving cars, human-computer interaction, and medical diagnosis and treatment. Given the exponential growth of digital devices, the potential for new applications seems unlimited.

Computer vision is challenging. Each image is a large, noisy, quantized, two-dimensional array of intensity values. Each intensity value is created by light reflected off a surface sitting in the three-dimensional world, focused by a lens, digitized and recorded by a camera. Information is lost and noise is added at each step. To address these challenges, computer vision research scientists and applications programmers are employing a wide variety of physical, engineering, mathematical, statistical, algorithmic and software techniques to develop computer vision systems.

Our goal in this course will be to learn about the challenges, the techniques, and the applications of computer vision. Since this is a computer science course, much of our focus will be on computational aspects of the computer vision problem. We will also consider potential impacts of advanced computer vision technologies on individuals and on society.

## Learning Objectives

At the end of this course, each successful student will be able to

- Apply techniques of calculus and linear algebra to solve problems involved in building the components of a computer vision system.
- Develop efficient algorithms for solving problems in computer vision.
- Write small-sized and intermediate-sized programs and also train neural networks to solve problems in computer vision.
- Map potential applications of computer vision into specific technical problems.
- Assess the difficulty of specific technical problems in computer vision and select potential solution techniques.
- Discuss thoughtfully some of the social implications of advanced computer vision technology.
- (6270 only) Evaluate the significance of the ideas and the thoroughness of the experimental analysis of a current research paper in the computer vision field.

## Prerequisites

Students should have had courses in programming, in data structures and in algorithms (e.g. CSCI 2300). Mathematical background should include a course in multivariable calculus and linear algebra (MATH 2100). This requirement is somewhat flexible since some students have done well in previous semesters without this background. In addition, we will be discussing the some of the necessary mathematical techniques as we proceed through the semester. Students may find my mathematical methods lectures notes helpful:

`http://www.cs.rpi.edu/~stewart/math_techniques`

## Requirements

Student grades will be determined by the following simple formula:

- 15% — lecture exercises and participation
- 85% — homework assignments

Letter grades will be determined based on the rounded, combined averages. Final cut-offs will be at the instructor's discretion, but will be no higher than 92 for an A, 89 for an A-, 86 for a B+, 82 for a B, etc. The same cutoffs will be used for the 4270 and 6270 courses. The grading and the curves will be different for each course due to differences in assignments.

## Lecture Exercises and Class Participation

Many lectures will be accompanied by short exercises to encourage engagement with the material prior to work on the more extensive homework problems. These will be managed through Submitty and graded gently. Class participation will involve a combination of asking questions during lecture and posting questions to Submitty's on-line chat. Students should actively be posting questions, comments, and suggestions. This includes threads that continue class discussions, that ask about applications, and that request help or clarification about homework problems. The goal is to create a learning community around the material of this course.

## Homework and Programming

Homework will involve solving mathematical problems, developing algorithms, writing programs, and analyzing results. The mixture of these will vary between assignments, but the programming aspect and resulting analysis will be the most important. Expect to write a lot of code.

Most homework assignments will be done individually. Occasionally, and only if explicitly allowed, some problems may be solved in teams of two.

Programming will be done using Python and OpenCV. Students will need at least the following Python packages:

- python, version 3.8.x
- numpy, version 1.19.x
- scipy, version 1.5.x
- matplotlib, version 3.3.x
- cv2, version 4.5.x

The dependencies between these can sometimes be a little tricky, so I recommend that you use a managed, isolated installation such as `anaconda`, together with the virtual environment it can create. See `https://www.anaconda.com/distribution`. You will have install OpenCV separately. To do this, find the anaconda bin directory (`/anaconda3` on my Mac) and type

```
pip install opencv-python
pip install opencv-contrib-python
```

You will eventually need both packages.

## Late Policy

Students have four free "late" days they can use on homework throughout the semester, with *at most two used for any one homework*. A late day is defined as any whole or partial day after the submission deadline. These free late days are to be used for minor illnesses, balancing other course work, problems with your computer, etc. Students do not need to use late days for substantial personal emergencies. Just get an excuse from your class dean and then together we can arrange a suitable time to complete any missed work.

## 4270 vs. 6270

There are two versions of this course, one at the 4000 level and one at the 6000 level, meeting together. The lecture material will be the same, but students in 6270 will need to complete more advanced work. This will be in the form of differences between problems in homework assignments, with more analytical questions assigned to students in 6270. There will be at least one problem — typically out of four to six — on each homework assignment that is required of students in 6270 but not students in 4270. In addition, students in 6270 will have one extra homework assignment requiring them to read and critically review a paper from the current research literature. Details of this assignment will be provided in the first month of the semester.

## Lectures, Lecture Notes and Resources

Lectures will be live via WebEx at an address distributed separately. These will be recorded and posted on-line for students to review later. Each lecture will be broken up into smaller segments with a break and Q/A discussion. I will try to post notes for each lecture on Submitty at least two days in advance of the class meeting.

While there is no textbook for the course, some references will be made to Rick Szeliski's book (`http://szeliski.org/Book/` — download the latest draft of the 2nd edition). General resources will be posted on the course Submitty site and pointers to reading material about each lecture will be embedded in the notes. Students are strongly encouraged to share references that they find helpful. The use of outside materials for this course is strongly encouraged and often a key to success.

## Academic Integrity

The Rensselaer Handbook of Student Rights and Responsibilities and The Rensselaer Graduate Student Supplement define various forms of Academic Dishonesty and procedures for responding to them. All forms are violations of the trust between students and teachers. Student-teacher relationships are built on trust. For example, students must trust that teachers have made appropriate decisions about the structure and content of the courses they teach, and teachers must trust that the assignments that students turn in are their own performance. Acts that violate this trust undermine the educational process.

The Rensselaer Handbook of Student Rights and Responsibilities and The Rensselaer Graduate Student Supplement define various forms of Academic Dishonesty and you should make yourself familiar with these. In this class, all assignments that are turned in for a grade must represent the student's own work. In cases where help was received, or teamwork was allowed, a notation on the assignment should indicate who you collaborated with. Submission of any assignment that is in violation of this policy will result in a penalty. If found in violation of the academic honesty policy, students may be subject to two types of penalty. The first violation will result in 0 grade for that assignment and up to an additional 5% overall grade penaly. The second violation will result in failure of the course. Electronic comparison tools will be used to find potential integrity violations. If you have any questions concerning this policy before submitting an assignment, please ask for clarification.

## Topic Schedule

Here is a summary of the topics we will be covering during the semester, along with tentative due dates for assignments. These will be updated throughout the semester.

| Lec. | Day | Date | Topic | Due |
|---|---|---|---|---|
| 1 | Mon. | 1/25 | Introduction; images | |
| 2 | Thu. | 1/28 | Numpy and OpenCV | |
| 3 | Mon. | 2/01 | Linear algebra | |
| 4 | Thu. | 2/04 | Lines and estimation | HW 1 |
| 5 | Mon. | 2/08 | Transformations | |
| 6 | Thu. | 2/11 | Image processing, part 1 | |
| | Mon. | 2/15 | NO CLASS | |
| 7 | Thu. | 2/18 | Image processing, part 2 | HW 2 |
| 8 | Mon. | 2/22 | Edge detection | |
| 9 | Thu. | 2/25 | Hough transforms | |
| 10 | Mon. | 3/01 | Interest points | |
| 11 | Thu. | 3/04 | Descriptors | HW 3 |
| 12 | Mon. | 3/08 | Camera geometry | |
| 13 | Thu. | 3/11 | Two image matching | |
| 14 | Mon. | 3/15 | Mosaics | |
| 15 | Thu. | 3/18 | Intro. to machine learning | HW 4 |
| 16 | Mon. | 3/22 | Neural network basics | |
| 17 | Thu. | 3/25 | Convolutional neural nets | |
| 18 | Mon. | 3/29 | Recognition, part 1 | |
| 19 | Thu. | 4/01 | Recognition, part 2 | HW 5 |
| 20 | Mon. | 4/05 | Data sets and evaluation | |
| 21 | Thu. | 4/08 | Detection | |
| 22 | Mon. | 4/12 | Segmentation, part 1 | |
| 23 | Thu. | 4/15 | Segmentation, part 2 | HW 6 |
| 24 | Mon. | 4/19 | Optical flow and tracking | |
| 25 | Thu. | 4/22 | Autonomous driving | |
| 26 | Mon. | 4/26 | Stereo | |
| 27 | Thu. | 4/29 | Face recognition | |
| 28 | Mon. | 5/03 | Conclusion | HW 7 |